IBIZ

อาจหาญ
สัตยารักษ์

เติมจินตนาการ
ตามสไตล์ของคุณ
ด้วยรูปกราฟิกและ
อักษรหลากหลายลีลา

THAI Banner

ชุดคอมพิวเตอร์ซอฟต์แวร์

THAI BANNER

อาจหาญ สัตยารักษ์

IBIZ

พิมพ์ครั้งที่ 1 พ.ศ. 2535

Thai Banner

โดย... **อาจหาญ สัตยารักษ์**

ISBN 974-88805-7-5

สงวนลิขสิทธิ์ตามพ.ร.บ.ลิขสิทธิ์ พ.ศ. 2521

ห้ามการลอกเลียนไม่ว่าส่วนหนึ่งส่วนใดของหนังสือเล่มนี้

นอกจากจะได้รับอนุญาต

จัดพิมพ์โดย

บริษัท อินฟอร์เมติก บิซิเนส พับลิเคชัน จำกัด

1420/1 ชั้น 3 อาคารศรีสุข พหลโยธิน 26

แขวงลาดยาว เขตบางเขน กรุงเทพฯ 10900

โทรศัพท์: 511-1146, 513-7637, 513-7649 และ 513-1833

โทรสาร: 511-1146

จัดจำหน่ายโดย

บริษัท ผู้จัดการ จำกัด

139-141 ถ.ริมคลองประปา แขวงบางซื่อ

เขตดุสิต กรุงเทพฯ 10800

โทรศัพท์: 587-8029-38

โทรสาร: 586-9110

พิมพ์ที่

บริษัท ด่านสุทธาการพิมพ์ จำกัด

239/11 หมู่ 8 ซ.จันทราสุข

ลาดพร้าว 87 กรุงเทพฯ

โทรศัพท์: 539-7490-4

โทรสาร: 539-2512

ราคา 290 บาท

ประวัติผู้เขียน

อาจหาญ สัตยารักษ์ ได้เข้ามามีส่วนเกี่ยวข้องกับคอมพิวเตอร์ โดยเริ่มมาจากการสนใจทางด้านคอมพิวเตอร์ตั้งแต่ปี พ.ศ.2526 โดยการศึกษาค้นคว้าด้วยตนเอง ในปี พ.ศ.2527 เริ่มเขียนบทความทางด้านคอมพิวเตอร์ลงตามนิตยสารต่าง ๆ ซึ่งส่วนใหญ่จะเป็นโปรแกรมทางด้านกราฟิก พ.ศ.2529 ชนะเลิศการประกวดโปรแกรมทางด้านการเรียนการสอนเรื่อง 'กฎการเคลื่อนที่ของนิวตัน' พ.ศ.2530 เคยไปช่วยงานที่สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี (สสวท.) ในสาขาอิเล็กทรอ-นิกส์และคอมพิวเตอร์ศึกษา เป็นเวลา 1 ปี เคยเป็นวิทยากรพิเศษด้านการเรียนกราฟิกบนเครื่องคอมพิวเตอร์ให้กับนักเรียนโรงเรียนสาธิตแห่งมหาวิทยาลัยเกษตรศาสตร์ เคยเป็นวิทยากรพิเศษด้านการเรียนกราฟิกและงานคอมพิวเตอร์ช่วยสอน (CAI) ให้กับนักศึกษาวิทยาลัยครูลำปาง และเคยเป็นที่ปรึกษาในการเขียนโปรแกรมด้าน CAI ให้แก่นิสิตปริญญาโทในสถาบันอุดมศึกษาอย่างไม่เป็นทางการ ปี พ.ศ. 2533 ได้พัฒนาโปรแกรม Thai Show ซึ่งเป็นโปรแกรมกราฟิกบนเครื่องพีซีและได้รับการยอมรับจากผู้ใช้อย่างกว้างขวาง ปัจจุบันเป็นอาจารย์สอนวิชาฟิสิกส์ระดับมัธยมปลาย ที่โรงเรียนลำปางกัลยาณี จ.ลำปาง และเป็นคณะกรรมการคอมพิวเตอร์ของกรมสามัญศึกษา ในส่วนของการผลิตและรวบรวมซอฟต์แวร์ด้านการเรียนการสอน (CAI) และคณะกรรมการผลิตซอฟต์แวร์ด้านการเรียนการสอนในรายวิชาพื้นฐานอาชีพคอมพิวเตอร์ของ สสวท.

คำนำ

ความรู้สึกเมื่อผู้เขียนเห็นโปรแกรมประเภท BANNER ของต่างประเทศนั้น จุดที่ผู้เขียนสนใจมากที่สุดอยู่ที่แนวทางการแสดงรูปแบบของตัวอักษรของโปรแกรมนั้น ซึ่งเมื่อเรามองอย่างผิวเผินจะดูประหนึ่งว่ามีอยู่หลายรูปแบบ แต่เมื่อพิจารณาอย่างลึกซึ้งแล้วจะพบว่าการแสดงรูปแบบของตัวอักษรนั้น ๆ ใช้เทคนิคในการสร้างอักษรประกอบการคำนวณจุดต่าง ๆ ในการแสดงผลเป็นประการสำคัญ

ประการแรกผู้เขียนถามตนเองว่า จะสามารถเขียนโปรแกรมในแนวนี้ด้วยตนเองได้หรือไม่ และด้วยความเชื่อมั่นในความสามารถของภาษาปาสคาลที่ผู้เขียนรู้อยู่เพียงน้อยนิดในขณะนั้น ผู้เขียนจึงตัดสินใจว่าจะลองเขียนมาใช้งานสักหนึ่งโปรแกรม

โปรแกรม Thai Banner จึงเกิดขึ้นด้วยเหตุนี้ โดยผู้เขียนมีจุดมุ่งหมายที่สำคัญคือ

1. ต้องสามารถสร้างตัวอักษรได้มากกว่า 4 ฟอนต์
2. แต่ละแบบต้องสามารถแสดงรูปแบบได้มากกว่า 8 แบบ
3. สามารถแสดงภาพประกอบกับข้อความได้
4. ต้องเขียนข้อความได้มากกว่า 20 ข้อความ
5. ในหนึ่งข้อความต้องสามารถใช้ฟอนต์ของตัวอักขระผสมกันได้มากกว่าหนึ่งฟอนต์
6. รูปแบบของป้ายประกาศจะต้องไม่มีรูปแบบตายตัว ผู้ใช้โปรแกรมสามารถออกแบบป้ายประกาศได้ตามความต้องการของตนเองเป็นประการสำคัญ

แนวคิดที่กำหนดไว้คือ โครงสร้างของอักขระแต่ละตัวจะต้องเก็บอยู่ในรูปของเวกเตอร์ เพราะเราสามารถนำไปขยายหรือวางเป็นเส้นที่กำหนดด้วยรูปร่างต่าง ๆ ได้ง่าย โดยโปรแกรมจะต้องอ่านโครงสร้างของอักขระทั้งหมดเข้ามาไว้ในหน่วยความจำได้ทั้งหมด เนื่องจากผู้เขียนต้องการให้สามารถใช้อักษรทั้งหมดในการสร้างป้ายประกาศหนึ่งป้ายได้

บริษัท อินฟอร์เมติก บิซิเนส พับลิเคชัน จำกัด (IBIZ) ได้มองเห็นประโยชน์ของโปรแกรมนี้ที่จะมีต่อวงการคอมพิวเตอร์ของไทย จึงได้เสนอให้ข้าพเจ้าเขียนคู่มือการใช้งานขึ้นมา พร้อมกับมีแนวทางของการพัฒนาโปรแกรมประกอบเพื่อนำเสนอต่อผู้อ่าน

หนังสือเล่มที่ท่านกำลังถืออยู่ในมือนี้จึงก่อให้เกิดประโยชน์ถึง 2 ประการคือด้วยกันคือ ใช้เป็นคู่มือของผู้ใช้ (user) โปรแกรม Thai Banner และช่วยให้โปรแกรมเมอร์ (programmer) ได้แนวทางในการพัฒนาโปรแกรม Thai Banner นี้ต่อไป

สำหรับโปรแกรม Thai Banner ที่เสนอนี้เป็นรุ่น 1.0 ใช้งานได้เฉพาะกับเครื่องพิมพ์แบบหัวเข็มเท่านั้น ผู้เขียนหวังว่าในรุ่นต่อไปจะพยายามพัฒนาเพื่อให้ใช้กับเครื่องพิมพ์แบบเลเซอร์ได้ด้วย

สุดท้ายนี้ผู้เขียนขอขอบคุณผู้ที่เกี่ยวข้องในการพัฒนาโปรแกรมนี้ทุก ๆ ท่าน ซึ่งได้แก่คณะครูอาจารย์โรงเรียนลำปางกัลยาณี ที่ช่วยเป็นผู้ทดลองใช้โปรแกรมนี้มาตั้งแต่ต้น ครอบครัวของผู้เขียนที่ให้เวลาบางส่วนมานั่งเขียนโปรแกรม พนักงานทุกคนของบริษัท อินฟอร์เมติก บิซิเนส พับลิเคชัน จำกัด (IBIZ) และท่านผู้อ่านทุกท่านที่ได้สละเงินซื้อหนังสือและโปรแกรมนี้ไปใช้งาน

ความบกพร่องของโปรแกรม Thai Banner นี้อาจจะมีอยู่บ้าง หากท่านพบกรุณาแจ้งข่าวให้แก่ผู้เขียนทราบด้วย จะเป็นพระคุณยิ่ง โดยท่านสามารถติดต่อไปได้ที่เลขที่ 222 ถ.พหลโยธิน ต.สวนดอก อ.เมือง จ.ลำปาง 52000 หรือจะกรุณาโทรศัพท์ไปที่หมายเลข 054-223091 เพื่อที่ผู้เขียนจะได้พัฒนาโปรแกรมนี้ให้มีความสมบูรณ์มากยิ่งขึ้นในรุ่นต่อ ๆ ไป

อาจหาญ สัตยารักษ์

สารบัญ

บทที่ 1 ไฟล์สำคัญของโปรแกรม Thai Banner..........11

- ปฐมเหตุของการพัฒนาโปรแกรม Thai Banner.....*12*
- ไฟล์สำคัญของโปรแกรม Thai Banner.....*12*
- การเข้าสู่โปรแกรม Thai Banner.....*14*

บทที่ 2 เมนูหลักและการเขียนป้ายประกาศ..........17

- เมนูหลัก.....*17*
- การเขียนป้ายประกาศ.....*18*
- การป้อนข้อความ.....*20*
- ขั้นตอนการป้อนข้อความ.....*22*
- การลงภาพ.....*29*

บทที่ 3 การแก้ไขข้อความและภาพ..........33

- การแก้ไขข้อความ.....*34*
- การแก้ไขภาพประกอบ.....*38*

❖ การลบข้อความเดิม.....41
❖ การเลือกรูปแบบพื้น.....41

บทที่ 4 การพิมพ์ป้ายประกาศ..45

❖ การดูป้ายประกาศบนจอภาพ.....46
❖ เมนูส่วนการพิมพ์.....47
❖ การพิมพ์ปกหนังสือ.....50
❖ การพิมพ์บนกระดาษที่กว้างกว่า 8 นิ้ว.....53

บทที่ 5 การอ่านและเขียนแฟ้มข้อมูลบนแผ่นบันทึก..55

❖ การเปลี่ยนสารบัญไฟล์.....57
❖ การเขียนแฟ้มข้อมูลโปรแกรม Thai Banner.....59
❖ การอ่านแฟ้มข้อมูลโปรแกรม Thai Banner.....61
❖ ข้อควรคำนึงในการอ่านและเขียนแฟ้มข้อมูล.....62

บทที่ 6 ความรู้เพิ่มเติมอื่น ๆ 65

❖ การสร้างภาพเพิ่มเติม.....65
❖ การติดตั้งลงฮาร์ดดิสก์.....66
❖ ข้อควรระวัง.....68

บทที่ 7 การสร้างฟอนต์แบบเวกเตอร์..71

❖ โครงสร้างของไฟล์ banner.fon.....71
❖ ยุทธวิธีการสร้างอักขระ.....72
❖ กำหนดตัวแปรใช้งาน.....74
❖ การสร้างรูปอักขระ.....78
❖ การเก็บอักขระลงไฟล์และการแสดงผลบนจอภาพ.....84
❖ โปรแกรมหลักการสร้างรูปอักขระ.....87
❖ ลำดับที่ของอักขระ.....93

บทที่ 8 การแสดงอักขระบนจอภาพ.....95

- การกำหนดตัวแปรใช้งาน.....*95*
- การคำนวณหาค่าตัวแปร.....*100*
- การแสดงอักขระบนจอภาพ.....*102*
- โปรแกรมหลักการแสดงอักขระ.....*105*

บทที่ 9 เทคนิคการแสดงข้อความ.....115

- การแสดงข้อความแบบขยายและตัวสูง 2 เท่า.....*115*
- การแสดงตัวเอนและตัวแบบโปร่ง.....*118*
- แสดงข้อความแบบเฉียงขึ้นลง.....*120*
- การแสดงข้อความแบบคลื่นและแนวโค้ง.....*123*
- ฝากไว้ให้คิด.....*127*

บทที่ 10 การสำเนาป้ายประกาศ.....129

- การพิมพ์ลงกระดาษ.....*130*
- การสำเนาจอภาพลงกระดาษ.....*131*
- เทคนิคการพิมพ์ป้ายประกาศ.....*136*
- วิเคราะห์การพิมพ์ป้ายประกาศ.....*140*
- การคำนวณหาจำนวนหน้าสูงสุด.....*141*

ภาคผนวก ก ตัวอย่างรูปในแฟ้มภาพที่ 1.....145

ภาคผนวก ข ตัวอย่างรูปในแฟ้มภาพที่ 2.....149

ไฟล์สำคัญของโปรแกรม Thai Banner

1

หนังสือเล่มนี้ผู้เขียนได้แบ่งเนื้อหาออกเป็น 2 ส่วน คือ

- ❖ ส่วนที่ 1 บทที่ 1-6 เป็นคู่มือการใช้โปรแกรม Thai Banner
- ❖ ส่วนที่ 2 บทที่ 7-10 เป็นแนวทางสำหรับให้โปรแกรมเมอร์ใช้เพื่อศึกษาเกี่ยวกับแนวทางการเขียนโปรแกรม Thai Banner

ดังนั้นถ้าท่านสนใจในส่วนใดก็อาจจะศึกษาเฉพาะส่วนนั้นได้ แต่ถ้าท่านเป็นผู้ที่สนใจในการเขียนโปรแกรมภาษาปาสคาลแล้ว ส่วนที่ 2 ของหนังสือเล่มนี้จะช่วยให้ท่านมีความรู้ในเรื่องการสร้างภาพกราฟิกได้เป็นอย่างดียิ่ง

ปฐมเหตุของการพัฒนาโปรแกรม Thai Banner

จากจุดที่ผู้เขียนเห็นโปรแกรมของต่างประเทศสามารถสร้างป้ายประกาศได้อย่างสวยงาม แต่ไม่สามารถใส่ภาษาไทยลงไปในโปรแกรมได้ ทั้งนี้เพราะว่าโปรแกรมมักยอมให้ป้อนข้อความได้เพียง 2 บรรทัด ซึ่งไม่พอสำหรับข้อความภาษาไทยที่ต้องใช้บรรทัดถึง 3 บรรทัด (ต่อข้อความภาษาไทยข้อความเดียว) เป็นอย่างน้อย

และสาเหตุที่สำคัญอีกประการหนึ่งที่ผู้เขียนประสบมาก็คือ ปัญหาในการไหว้วานให้ครูศิลปะมาช่วยในการเขียนป้ายรายชื่อวิทยากรในการประชุมแต่ละครั้ง (เพราะเขามักมีงานประจำมากอยู่แล้ว) จึงคิดว่าคอมพิวเตอร์น่าจะช่วยเราได้

แต่เครื่องคอมพิวเตอร์ไม่ใช่เครื่องมือที่วิเศษเหมือนที่คนบางกลุ่มคิด เพราะว่าการไปนั่งเล่าความต้องการของเราให้เครื่องคอมพิวเตอร์ฟังแล้วเครื่องก็พัฒนาโปรแกรมมาให้ตามต้องการนั้นมันเป็นเรื่องที่เป็นไปไม่ได้ เราจึงจะต้องคิดเองเขียนเองแล้วนำสิ่งนั้นผ่านภาษาคอมพิวเตอร์ที่เราถนัดจึงจะทำให้ได้งานตามต้องการ ผู้เขียนจึงต้องมากำหนดแนวคิดและเขียนโปรแกรมขึ้นมาใช้งาน โดยผู้เขียนใช้ภาษาปาสคาลของเทอร์โบปาสคาลรุ่น 5.5 มาช่วยในการพัฒนา

ไฟล์สำคัญของโปรแกรม Thai Banner

สำหรับแผ่นโปรแกรม Thai Banner ที่ขายพร้อมหนังสือเล่มนี้ ถูกบรรจุลงในแผ่นดิสก์ขนาด 360 KB ซึ่งประกอบด้วยระบบปฏิบัติการของดอสและไฟล์อีก 10 ไฟล์ดังนี้

1. AUTOEXEC.BAT	33	ไบต์
2. TBANNER1.FON	10458	ไบต์
3. TBANNER2.FON	9375	ไบต์
4. TBANNER3.FON	7745	ไบต์
5. TBANNER4.FON	9397	ไบต์

6.	TBANNER1.PIC	66528	ไบต์ (108 ภาพ)
7.	TBANNER2.PIC	59136	ไบต์ (96 ภาพ)
8.	TBANNER.EXE	46825	ไบต์
9.	HERC.BGI	6125	ไบต์
10.	EGAVGA.BGI	5363	ไบต์

โดยมีรายละเอียดของแต่ละไฟล์ดังนี้

❖ ไฟล์ autoexec.bat เป็นไฟล์ที่ใช้สำหรับรันไฟล์ tbanner.exe โดยอัตโนมัติเมื่อทำการใส่แผ่นโปรแกรม Thai Banner นี้ที่ไดรฟ์ A แล้วเปิดเครื่อง

❖ ไฟล์ tbanner1.fon-tbanner4.fon เป็นไฟล์ที่เก็บโครงสร้างแบบเวกเตอร์ของอักษรภาษาไทยและภาษาอังกฤษที่ใช้ในโปรแกรม Thai Banner ซึ่งจะมีขนาดของแต่ละอักษรไม่เท่ากัน โดยไฟล์ทั้งสี่ไฟล์นี้จะถูกเรียกเข้ามาเก็บไว้ในหน่วยความจำของเครื่องคอมพิวเตอร์ในทันทีที่รันไฟล์ tbanner.exe

❖ ไฟล์ tbanner1.pic-tbanner2.pic เป็นแฟ้มภาพที่ผู้เขียนดัดแปลงมาจากแฟ้มภาพของโปรแกรมต่างประเทศ แต่มีโครงสร้างของภาพที่แตกต่างจากต้นฉบับเดิม จึงจะนำไปใช้ในโปรแกรมต้นฉบับเดิมไม่ได้ ในทำนองเดียวกันก็จะนำแฟ้มภาพของเดิมมารันกับโปรแกรม Thai Banner ไม่ได้เช่นกัน ผู้อ่านสามารถสร้างภาพใหม่เพิ่มเติมได้โดยการใช้ไฟล์ editpic.exe ของโปรแกรมกราฟิกไทยโชว์ ซึ่งเป็นระบบงานกราฟิกอีกเล่มหนึ่งที่ผู้เขียนพัฒนาขึ้นมาสร้างภาพเพิ่มเติมได้ (จำนวนภาพในแฟ้มภาพมี 108 ภาพ และ 96 ภาพตามลำดับ)

❖ ไฟล์ tbanner.exe เป็นไฟล์หลักในการเขียนหรือพิมพ์ป้ายประกาศ ไฟล์นี้ผู้เขียนทำให้มีขนาดเล็กลงได้ด้วยไฟล์ pklite.exe แต่ถ้า

ต้องการจะรันด้วยไฟล์ที่เป็นขนาดเดิมก็หาไฟล์ pklite.exe มารันเพื่อคืนสภาพได้ด้วยการใช้คำสั่งดังนี้

```
A>pklite tbanner -x
```

❖ ไฟล์ herc.bgi และ egavga.bgi คือไฟล์สำหรับจัดกราฟิกให้เหมาะสมกับจอภาพที่ท่านใช้คือ จอโมโนโครม จอสี จอวีจีเอโมโนโครม หากท่านใช้จอภาพชนิดอื่นก็ให้หาไฟล์ *.bgi ตามชนิดของจอใส่ในแผ่น (ต้องใช้ไฟล์ *.bgi ที่ใช้กับเทอร์โบปาสคาล 5.5)

❖ เมื่อมีการสร้างป้ายประกาศและเก็บลงในแผ่นบันทึก จะมีชื่อสกุล BAN ทุกครั้ง โดยโปรแกรม Thai Banner จะใส่ชื่อสกุลให้เองโดยอัตโนมัติ ทั้งในช่วงการเก็บและการอ่านแฟ้มข้อมูล ซึ่งแต่ละแฟ้มของป้ายประกาศจะแยกเป็นเอกเทศจากกัน

การเข้าสู่โปรแกรม Thai Banner

เนื่องจากโปรแกรม Thai Banner เป็นโปรแกรมที่ทำงานด้วยตนเองทั้งหมด โดยไม่ได้อ้างอิงส่วนต่าง ๆ ของระบบปฏิบัติการของดอสรุ่นใด ๆ จึงสามารถใช้งานกับดอสได้ทุกรุ่น ทั้งยังสามารถใช้งานได้กับเครื่องที่มีฮาร์ดดิสก์ได้อย่างไม่มีเงื่อนไขใด ๆ เช่นกัน

การรันไฟล์ของโปรแกรม Thai Banner จึงทำง่าย ๆ ได้ดังนี้

❖ กรณีที่ยังไม่ได้เปิดเครื่อง ก็ใส่แผ่นโปรแกรม Thai Banner ในไดรฟ์ A แล้วเปิดเครื่องคอมพิวเตอร์ จะเป็นการเข้าสู่โปรแกรม Thai Banner ด้วยไฟล์ชื่อ autoexec.bat ทันที

❖ กรณีที่อยู่ในดอส เช่น ใน A:\> จะสามารถรันไฟล์ tbanner.exe ได้เลย โดยการป้อนผ่านดอสดังนี้

A:\>tbanner ตามด้วยการกดแป้น [Enter]

❖ ถ้าท่านใช้งานกับฮาร์ดดิสก์ก็คงทำได้ดังนี้

1. สร้างไดเรกทอรีย่อยชื่อ TBANNER (หรืออาจจะให้ชื่ออื่นก็ได้)

C:\>md tbanner

2. เข้าสู่ไดเรกทอรีย่อยที่สร้าง

C:\>cd tbanner

3. สำเนาไฟล์ของโปรแกรม Thai Banner จากไดรฟ์ A มาเก็บไว้ในไดเรกทอรีที่สร้าง

C:\TBANNER>copy a:*.*

4. ลบไฟล์ command.com ออกไปเพราะว่าไม่มีความจำเป็นต้องใช้งานอีกแล้ว (อาจลบไฟล์ autoexec.bat ทิ้งด้วยก็ได้)

C:\TBANNER>del command.com

❖ ท่านก็จะได้ไฟล์ของโปรแกรม Thai Banner ที่สมบูรณ์และพร้อมที่จะใช้งานกับฮาร์ดดิสก์ได้แล้ว

5. สามารถเรียกใช้โปรแกรมนี้ได้โดยเรียกไฟล์ tbanner.exe ซึ่งจะเรียกใช้ในขณะที่อยู่ในไดเรกทอรีใด ๆ ก็ได้ เช่น

C:\tbanner

C:\DOS\tbanner

เมื่อท่านได้ทำงานตามขั้นตอนดังกล่าวข้างต้นเรียบร้อยแล้ว รอสักครู่ ไฟล์ tbanner.exe จะทำการโหลดอักขระที่ใช้งานมาสู่หน่วยความจำ แล้วแสดงให้ทราบว่า ท่านกำลังอยู่ในโปรแกรม Thai Banner รุ่น 1.0 ดังรูปที่ 1.1

รูปที่ 1.1 *จอภาพแรกเมื่อเข้าสู่โปรแกรม Thai Banner*

เมนูหลักและการเขียนป้ายประกาศ 2

โปรแกรม Thai Banner จะอ่านข้อมูลของอักขระทั้งสองมาสู่หน่วยความจำของเครื่องคอมพิวเตอร์โดยอัตโนมัติ แต่เนื่องจากไฟล์อักขระทั้งหมดนี้ได้ออกแบบให้ใช้หน่วยความจำส่วนที่เรียกว่า ฮีป (heap) ในการใช้งาน จึงจะต้องมีการตรวจสอบว่า เครื่องที่ใช้งานนั้นมีหน่วยความจำเพียงพอหรือไม่ ถ้าไม่เพียงพอก็จะมีข้อความแจ้งให้ทราบว่าเกิดความผิดพลาด หน่วยความจำไม่พอรันโปรแกรมและจะกลับไปสู่ดอส ซึ่งก็แสดงว่าเครื่องของท่านใช้โปรแกรม Thai Banner นี้ไม่ได้

เมนูหลัก เมื่อโปรแกรมได้ดำเนินการอ่านไฟล์ข้อมูลอักขระจำนวน 4 ฟอนต์เรียบร้อยแล้ว จะเข้าสู่ส่วนของเมนูหลักโปรแกรม Thai Banner โดยมีรายการเลือกดังนี้

- ❖ การเขียนป้ายประกาศ (ภาพกระดาษ)
- ❖ การดูผลของป้ายประกาศบนจอภาพ (ภาพเครื่องคอมพิวเตอร์)
- ❖ การพิมพ์ป้ายประกาศลงกระดาษ (ภาพเครื่องพิมพ์)
- ❖ การอ่าน-เขียนข้อมูลป้ายประกาศลงแผ่นบันทึก (ภาพแผ่นบันทึก)
- ❖ เลิกงาน (ภาพอักขระ A หรือ B หรือ C ฯลฯ)

รวมส่วนของเมนูหลักทั้งหมด 5 รายการ และเพื่อให้ส่วนของเมนูหลักนี้มีความแปลกตาออกไป จึงได้ออกแบบให้เป็นรูปภาพแทน ซึ่งแต่ละรูปภาพจะสื่อความหมายให้ทราบว่าเป็นส่วนการทำงานในเรื่องใด ในเมนูหลักนี้จะเป็นการเลือกการทำงานแบบเลื่อนแถบสว่างโดยการกดแป้นลูกศรซ้ายขวา ซึ่งโปรแกรมจะแสดงรายการของแต่ละตัวเลือกให้ทราบทุกครั้งที่เลื่อนแถบสว่างมาอยู่ ณ รายการเลือกนั้น ๆ จึงทำให้ทราบว่าแต่ละตัวเลือกนั้นทำงานในเรื่องใดได้บ้าง ดังจะเห็นได้จากในรูปที่ 2.1

รูปที่ 2.1 *โปรแกรมหลักของโปรแกรม Thai Banner*

การเขียนป้ายประกาศ

เมื่อเราเลื่อนแถบสว่างมาอยู่ที่การเขียนป้ายประกาศ พร้อมกับกดแป้น [Enter] เพื่อเลือกการทำงานตามรายการนี้ จะมีการแสดงรายการย่อยเพื่อให้เลือกการทำงานได้ดังนี้

- ***ป้อนข้อความ*** เพื่อป้อนข้อความในการทำป้ายประกาศ โดยสามารถจะป้อนได้สูงสุดถึง 20 ข้อความ (มากกว่าของโปรแกรมต่างประเทศหลายเท่า)

- ***แก้ไขข้อความ*** สำหรับใช้แก้ไขข้อความที่ได้เขียนไปในตอนแรก เพื่อให้ได้ข้อความที่เปลี่ยนไป หรือให้ได้รูปแบบการแสดงผลที่แตกต่างออกไป

- ***การลงภาพ*** เป็นการเลือกภาพที่มีให้มาแล้วประกอบร่วมเข้าไปในป้ายประกาศที่ได้เขียนขึ้นมา ซึ่งได้กำหนดจำนวนภาพสูงสุดต่อหนึ่งป้ายประกาศเป็น 20 ภาพ โดยเลือกขนาดได้ 2 ขนาดคือ ขนาดเล็กและขนาดโต 2 เท่า

- ***เลือกรูปแบบพื้น*** ในกรณีที่ต้องการให้พื้นของกระดาษมีลวดลายแปลกตามากกว่าปล่อยให้เรียบ ๆ ตามปกติ ก็สามารถกำหนดรูปแบบพื้นของป้ายประกาศทั้งหมดได้จากรายการย่อยนี้

- ***ลบข้อความเดิม*** สำหรับรายการนี้เป็นการลบป้ายประกาศของเดิมทั้งแผ่นป้ายในหน่วยความจำออกไป กล่าวคือเป็นการตั้งต้นเขียนป้ายประกาศแผ่นใหม่นั่นเอง

- ***กลับเมนูหลัก*** เพื่อกลับคืนสู่เมนูหลักในกรณีที่เรายุติการเขียนป้ายประกาศแผ่นนั้น หรือต้องการจะออกไปทำงานในเมนูหลักอื่น ๆ

การป้อนข้อความ

เมื่อเราได้เลือกรายการย่อยที่ 1 ซึ่งก็คือการป้อนข้อความ จะปรากฏส่วนของการป้อนข้อความดังรูปที่ 2.2

รูปที่ 2.2 *ส่วนการป้อนข้อความและแป้นที่ใช้ควบคุม*

ส่วนสำคัญของการป้อนข้อความที่แสดงบนจอภาพมีดังนี้

❖ แสดงให้ผู้ใช้ทราบว่ากำลังป้อนข้อความลำดับที่เท่าใดในจำนวนข้อความทั้งหมด 20 ข้อความ ถ้ามีการป้อนข้อความครบ 20 ข้อความแล้วจะมีข้อความแจ้งให้ทราบว่า *มีข้อความครบ 20 ข้อความแล้ว* จึงไม่สามารถป้อนข้อความต่อไปได้อีก และโปรแกรมจะย้อนกลับไปที่เมนูย่อยตามเดิม

❖ ข้อความที่แสดงการใช้แป้นพิเศษจะปรากฏในที่ว่างของเมนูหลัก โดยมีแป้นที่ใช้งานดังต่อไปนี้

[^]-[B] เข้า-ออกตัวสูง 2 เท่า

[^]-[L] เข้า-ออกตัวเอียง

[^]-[D] กำหนดตัวนูนด้านล่าง

[^]-[V] เข้า-ออกตัวห้อยท้าย

[^]-[P] ตัวเงาบน

[^]-[E] เข้า-ออกตัวขยาย

[^]-[S] กำหนดตัวนูนด้านบน

[^]-[T] เข้า-ออกตัวยกกำลัง

[^]-[O] ตัวเงาล่าง

[F1]-[F4] เลือกฟอนต์ตัวอักขระได้ 4 แบบ โดยขณะที่กำลังทำการป้อนข้อความนี้ จะไม่เห็นความเปลี่ยนแปลงใด ๆ ของข้อความบนจอภาพ

[F10] ดูรูปแบบของข้อความที่ปรากฏจริงบนป้ายประกาศ ซึ่งโปรแกรมจะแสดงส่วนของข้อความที่กำลังป้อนให้เห็นที่กราฟิกอีกหน้าหนึ่ง เพราะกราฟิกที่ใช้งานนี้มีอยู่ 2 หน้า โดยหน้าแรกจะใช้เป็นส่วนของการป้อนข้อความ และอีกหน้าหนึ่งไว้สำหรับแสดงข้อความที่เป็นจริงเมื่อสร้างป้ายประกาศ (เพียงแต่จะมีขนาดเล็กที่สุดเท่านั้น) หากข้อความที่ป้อนมีความยาวมาก โปรแกรมจะแสดงส่วนการดูข้อความนี้หลายบรรทัด ซึ่งจะทำให้ได้ข้อความสมบูรณ์เท่าที่ได้ป้อนไป

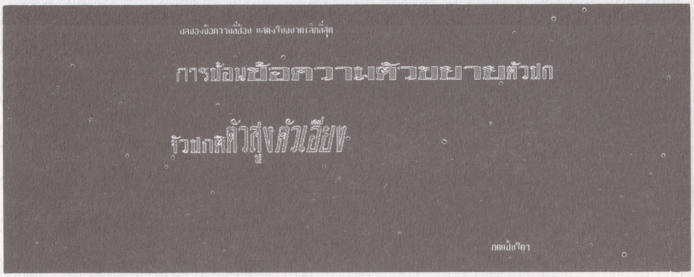
ตราบเท่าที่ยังไม่มีการกดแป้น [Enter] ในการป้อนข้อความ ท่านจะสามารถกดแป้น [F10] มาดูข้อความที่เป็นลักษณะของป้ายประกาศนี้ได้ทุกเวลา

เมื่อ [^]-[B] คือการกดแป้น [Ctrl] และแป้น [B] พร้อมกัน โปรดสังเกตว่ารหัสที่ใช้นี้ไม่ต้องกดแป้น [Ctrl]+[P] ก่อน แต่เป็นการกดแป้นเพียงชุดเดียวเท่านั้น สำหรับ [^]-[D], [^]-[S], [^]-[O] และ [^]-[P] นั้น จะทำงานอย่างใดอย่างหนึ่งเท่านั้น นั่นคือ ถ้าเข้า [^]-[S] ไว้ต่อมาป้อน [^]-[D] ก็หมายความว่ายกเลิก [^]-[S] มาใช้ [^]-[D] แทน

โปรแกรมจะแสดงรหัสควบคุมการแสดงตัวข้อความตามที่กล่าวมาทั้งหมดนี้ด้วยการแสดงเป็นข้อความแบบกลับขาวดำที่ส่วนบนและล่างของข้อความ ถ้ามีการป้อนรหัสควบคุมนี้ซ้ำในที่เดิมก็อาจจะมองไม่เห็นส่วนรหัสเดิมที่ลงไปก่อนหน้า แต่ถ้ากดแป้น [F10] ก็พอจะพิจารณาได้ว่ามีรหัสควบคุมอยู่ที่ตำแหน่งใดบ้าง

ตำแหน่งภาษาไทยของแป้นพิมพ์ใช้แบบที่ใช้กันอยู่โดยทั่ว ๆ ไป ทั้งนี้รวมทั้งเลขไทยด้วย เช่น เลข ๑ จะอยู่ที่ [Shift+2] เป็นต้น

ตัวอย่างของข้อความที่ป้อนในรูปที่ 2.2 เมื่อกด [F10] จะเป็นดังรูปที่ 2.3



รูปที่ 2.3 *ข้อความที่ป้อนในรูปที่ 2.2 เมื่อดูรูปแบบด้วยแป้น [F10]*

ขั้นตอนการป้อนข้อความ

สำหรับขั้นตอนในการป้อนข้อความจะเป็นไปตามลำดับดังต่อไปนี้

- ❖ ป้อนข้อความโดยใช้รหัสควบคุมตามที่กล่าวมาแล้วได้ถ้าท่านต้องการให้ป้ายประกาศมีรูปแบบตามการกำหนดของผู้เขียนเอง สำหรับข้อความเหล่านี้เรายังสามารถแก้ไขได้อีก หากดูบนจอภาพตอนสร้างป้ายประกาศแล้วยังไม่พอใจ
- ❖ เมื่อกดแป้น [Enter] ซึ่งหมายถึงการสิ้นสุดการป้อนข้อความนั้น ๆ ก็จะเป็นการเลือกขนาดของข้อความที่จะนำไปเป็นป้ายประกาศ โดย

โปรแกรมจะแสดงตัวอย่างของอักขระโดยการสุ่มค่าขึ้นมาหนึ่งตัว ซึ่งจะแสดงอยู่ในส่วนของเมนูหลักเช่นเดิม สำหรับแป้นที่ใช้ในการเลือกขนาดก็คือแป้นลูกศรทั้งสี่แป้น โปรแกรมจะแสดงอักขระนั้น ๆ ในขนาดที่เปลี่ยนไปแบบวนรอบ

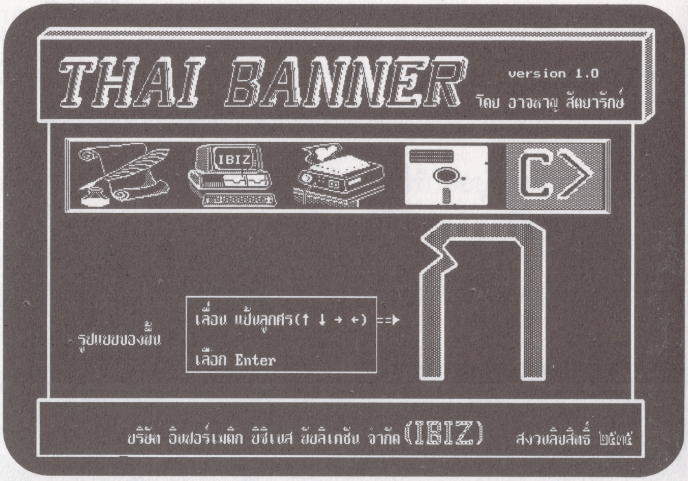
❖ ให้เลื่อนแป้นลูกศรจนได้ขนาดที่ท่านต้องการ แล้วให้กดแป้น [Enter] ตัวอย่างของการเลือกขนาดจะเป็นดังรูปที่ 2.4

รูปที่ 2.4 *ส่วนการเลือกขนาดของข้อความ*

❖ เมื่อได้เลือกขนาดของข้อความแล้ว ขั้นต่อไปก็คือการเลือกรูปแบบพื้นของข้อความที่เราต้องการ โดยจะมีรูปแบบของพื้นให้เลือกอยู่ 21 แบบ โดยการใช้แป้นลูกศรทั้งสี่ก็จะเปลี่ยนอักขระที่แสดงบนจอภาพให้เห็นรูปแบบพื้นที่เลือก

เฉพาะตัวอักขระขนาดเล็กอาจมองเห็นรูปแบบของพื้นยากนิดหนึ่ง แต่ก็พอจะมองเห็นรูปแบบอย่างคร่าว ๆ ได้

ตัวอย่างของการเลือกรูปแบบพื้นจะแสดงบนจอภาพได้ดังรูปที่ 2.5



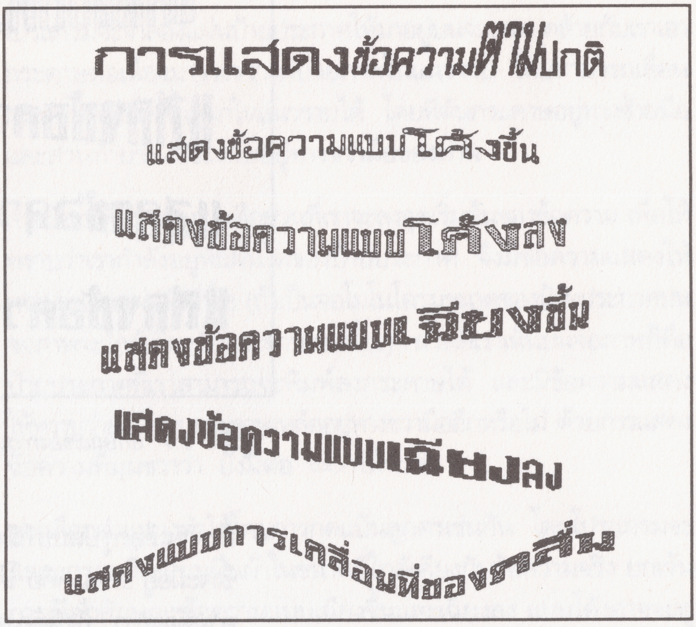
รูปที่ 2.5 *การเลือกรูปแบบพื้นของตัวอักขระในข้อความ*

❖ หลังจากที่เลือกขนาดและรูปแบบของพื้นแล้วก็จะมายังส่วนของการเลือกลักษณะการแสดงข้อความบนป้ายประกาศ ซึ่งท่านจะต้องเลือกว่าต้องการรูปแบบเช่นใด ก็สามารถทำได้โดยการกดแป้นลูกศรเช่นกัน ซึ่งจะมีรูปแบบให้เลือกทั้งสิ้น 12 รูปแบบ คือ

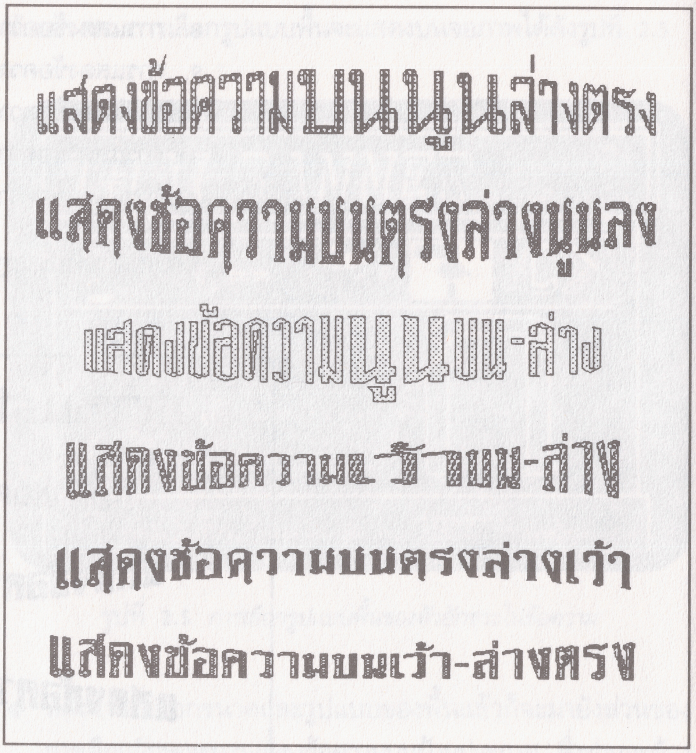
1. การแสดงข้อความตามปกติ
2. การแสดงข้อความแบบโค้งขึ้น
3. การแสดงข้อความแบบโค้งลง
4. การแสดงข้อความแบบเฉียงขึ้น
5. การแสดงข้อความแบบเฉียงลง
6. การแสดงข้อความแบบคลื่น
7. การแสดงข้อความแบบบนนูนล่างตรง

8. การแสดงข้อความแบบบนตรงล่างนูน
9. การแสดงข้อความแบบนูนบนและล่าง
10. การแสดงข้อความแบบเว้าบนและล่าง
11. การแสดงข้อความแบบบนตรงล่างเว้า
12. การแสดงข้อความแบบบนเว้าล่างตรง

โดยรูปแบบที่ปรากฏบนจอภาพจะเป็นดังรูปที่ 2.6 และ 2.7



รูปที่ 2.6 *ลักษณะของการแสดงข้อความ 6 แบบแรก (ข้อ 1 - ข้อ 6)*



รูปที่ 2.7 *ลักษณะของการแสดงข้อความ 6 แบบหลัง (ข้อ 7 - ข้อ 12)*

❖ ถ้าท่านเลือกรูปแบบที่มีความโค้ง โปรแกรมจะให้เลือกความโค้ง ซึ่งจะมีอยู่ 3 ขนาดคือ โค้งน้อย โค้งปานกลาง และโค้งมาก ให้เลือกตามต้องการ ทั้งนี้ต้องคำนึงถึงความยาวของข้อความด้วย

สำหรับการเลือกลักษณะของข้อความแบบโค้งนี้ หากข้อความที่กำหนดนี้ยาวเกินไป เมื่อโปรแกรมคำนวณแล้วพบว่าไม่สามารถเขียนได้ก็จะเปลี่ยนลักษณะการแสดงข้อความที่เลือกไว้ไปเป็นการแสดงแบบปกติแทน (โดยไม่มีข้อความแจ้งให้ทราบ)

นอกจากนี้ท่านต้องทราบด้วยว่าการเลือกลักษณะข้อความที่มีความโค้งนี้ เมื่อแสดงผลเป็นป้ายประกาศจะมีขนาดโตกว่าขนาดจริงที่เลือกไว้เสมอ โดยตัวที่มีขนาดเล็กที่สุดจะใกล้เคียงกับขนาดที่เลือก

ดังนั้นหากข้อความที่แสดงลักษณะไม่ตรงกับที่ท่านเลือกไว้ ก็ขอให้เข้าใจว่าไม่ได้เกิดความผิดพลาดขึ้นในโปรแกรมแต่อย่างใด

❖ ขั้นตอนสุดท้ายของการเขียนข้อความก็คือ การเลือกตำแหน่งที่จะเขียนข้อความในป้ายประกาศที่สร้าง โดยมีแนวคิดดังนี้

โปรแกรมจะจำลองแผ่นป้ายประกาศให้มาอยู่บนจอภาพ คล้ายกับเราเอากระดาษต่อเนื่องมาวางขวางกับจอภาพในแนวราบ โดยสามารถเลื่อนกระดาษต่อเนื่องนี้ไปมาในแนวราบได้ โดยที่ต้นกระดาษอยู่ทางซ้ายมือและส่วนท้ายของกระดาษอยู่ทางขวามือของท่าน

ส่วนที่เราเห็นบนจอภาพคือส่วนที่เราจะลงจุดเริ่มต้นของข้อความ เพื่อให้ทราบว่าเรากำลังอยู่ที่แผ่นใดของป้ายประกาศ จึงมีข้อความแสดงให้ทราบว่าเป็นแผ่นที่เท่าใด (ถ้าเป็นจอโมโนโครมขนาดของป้ายประกาศบนจอภาพจะใกล้เคียงกับกระดาษต่อเนื่อง) ส่วนที่เราเห็นบนจอภาพก็คือป้ายประกาศที่เราสามารถจะพิมพ์ลงกระดาษได้ และมีข้อความแสดงให้ทราบว่ายังมีแผ่นกระดาษเหลืออยู่ทางขวามืออีกหรือไม่ ด้วยการแสดงข้อความที่มุมขวาว่า ยังมีต่อ ==> ให้ทราบ

การเลือกตำแหน่งทำได้โดยการกดแป้นลูกศรเช่นกัน โดยโปรแกรมจะแสดงกรอบสี่เหลี่ยมผืนผ้าในขนาดที่ใกล้เคียงกับข้อความจริง ยกเว้นการเลือกลักษณะข้อความแบบเฉียงขึ้นและเฉียงลง แบบโค้งอาจจะดูขอบเขตของข้อความลำบากนิดหนึ่ง แต่ที่ขอบซ้ายขวาจะตรงกับความจริง

ท่านสามารถเลื่อนข้อความไปยังส่วนต่างๆของป้ายประกาศได้ โดยจะเป็นการเลื่อนแบบโปรแกรมพิมพ์เอกสารทั่ว ๆ ไป คือสามารถเลื่อน

ได้ทั้งทางซ้ายขวาและเลื่อนขึ้นลง จึงสามารถนำข้อความที่ป้อนไปแสดงที่ส่วนใดของป้ายประกาศก็ได้

แป้น [F8] ใช้เพื่อกำหนดระยะเลื่อนข้อความให้มีความไวมากขึ้นหรือน้อยลง

แป้น [F10] ใช้เพื่อทดสอบว่าข้อความที่ปรากฏบนจอภาพตรงกับความต้องการหรือไม

รูปที่ 2.8 เป็นตัวอย่างของการแสดงรูปสี่เหลี่ยมที่เป็นตัวแทนของข้อความ

รูปที่ 2.8 *แสดงการเลือกตำแหน่งสำหรับการแสดงข้อความ*

เมื่อเราเลื่อนเคอร์เซอร์ (กรอบสี่เหลี่ยมที่แทนข้อความหรือกรอบสี่เหลี่ยมที่แทนภาพ) ไปยังด้านขวาสุดของจอภาพก็จะข้ามไปยังป้ายประกาศในหน้าถัดไป (คล้ายกับเราดึงกระดาษต่อเนื่องไปทางซ้าย) หากเลื่อนเคอร์เซอร์ไปยังด้านซ้ายสุดของจอภาพก็จะตรงไปยังแผ่นที่อยู่ก่อนหน้านี้ เช่น เมื่อเลื่อนแผ่นที่ 3 ก็จะไปยังแผ่นที่ 2 เป็นต้น (ซึ่งจะคล้ายกับการที่เราดึงกระดาษต่อเนื่องไปทางขวามือ)

ยกเว้นเมื่อเลื่อนไปถึงแผ่นแรกหรือถึงแผ่นสุดท้าย ก็จะไม่มีการเลื่อนไปยังหน้าอื่น (ตามเงื่อนไขที่กล่าวมาข้างต้นทั้งหมด)

เมื่อเราได้ตำแหน่งที่ต้องการแล้วให้กดแป้น [F10] เพื่อดูผลการลงข้อความบนป้ายประกาศเสียก่อน ถ้าข้อความลงในแผ่นที่เลือกไม่หมดก็สามารถกดแป้นใดก็ได้ (ยกเว้นแป้น [Enter]) เพื่อดูการลงข้อความในแผ่นต่อไปว่าเป็นอย่างไร

กดแป้น [Esc] จะกลับไปยังส่วนการเลือกตำแหน่งแสดงข้อความใหม่อีกครั้งหนึ่ง

ถ้าเห็นว่าข้อความแสดงในตำแหน่งที่ต้องการ ก็ให้กดแป้น [Enter] ก็จะจบการป้อนข้อความในบรรทัดนั้นและกลับเข้าสู่เมนูหลักเพื่อทำงานในเรื่องอื่นต่อไป

การลงภาพ

โปรแกรม Thai Banner นี้สามารถลงภาพประกอบได้โดยอาศัยภาพที่มีอยู่ในแผ่นโปรแกรมจำนวน 2 แฟ้มภาพ (เคยกล่าวถึงแล้วในบทที่ 1) โดยมีภาพทั้งหมด 108 และ 96 ภาพตามลำดับ

ผู้เขียนป้ายประกาศสามารถกำหนดขนาดของภาพได้ 2 ขนาดคือ ขนาดเล็กและขนาดโต 2 เท่า โดยจะมีข้อความแสดงแนวทางการเลือกให้ขั้นตอนในการลงภาพจะเป็นดังนี้คือ

- ❖ เลื่อนแถบสว่างของเมนูหลักมาที่รูปปากกากับกระดาษแล้วกด [Enter] จะเกิดเมนูย่อยให้เลือกการทำงาน
- ❖ เลื่อนแถบสว่างของเมนูย่อยโดยแป้นลูกศรขึ้นลงให้มาอยู่ที่ข้อความว่า การลงภาพ แล้วกดแป้น [Enter]
- ❖ ตอนนี้จะปรากฏข้อความให้เลือกว่าจะอ่านจากแฟ้มภาพที่ 1 หรือแฟ้มภาพที่ 2 ให้เลื่อนแถบสว่างมายังส่วนที่ต้องการแล้วกดแป้น [Enter]

- ❖ จะมีเมนูย่อยให้เลือกซ้ำอีกครั้งว่าจะเลือกขนาดใหญ่หรือขนาดเล็ก ให้เลื่อนแถบสว่างมายังขนาดที่ท่านต้องการแล้วกดแป้น [Enter]

ซึ่งรายการที่กล่าวมาทั้งหมดนี้จะปรากฏผลบนจอภาพ ดังรูปที่ 2.9

รูปที่ 2.9 *ส่วนของการเลือกแฟ้มภาพและขนาดของภาพ*

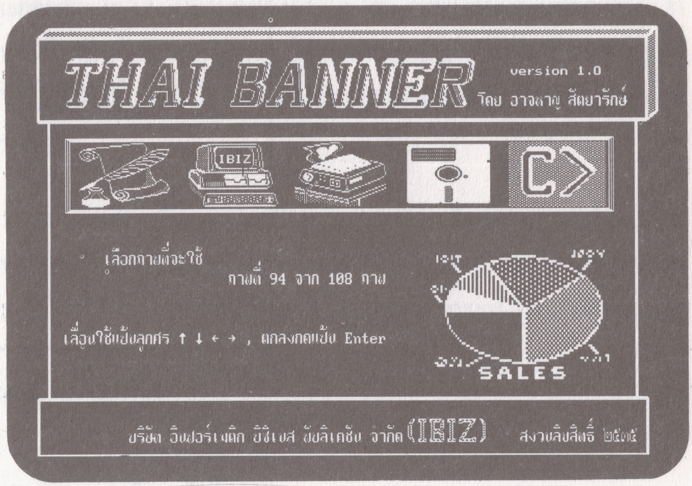
ในการเลือกภาพจากแฟ้มภาพนี้ท่านจะต้องมีไฟล์ tbanner1.pic และ tbanner2.pic อยู่ในไดเรกทอรีที่เรียกใช้ไฟล์ tbanner.exe ด้วยเสมอ เช่น เมื่อรันไฟล์ tbanner.exe ที่ไดรฟ์ A ก็จะต้องมีแฟ้มภาพทั้งสองแฟ้มอยู่ในแผ่นบันทึกที่ไดรฟ์ A ด้วย ดังนั้นการรันโปรแกรม Thai Banner ผ่านฮาร์ดดิสก์จึงใช้งานได้คล่องตัวกว่า

สำหรับภาพที่มีในป้ายประกาศนี้ผู้เขียนได้ดัดแปลงมาจากซอฟต์แวร์ของต่างประเทศ โดยใช้โครงสร้างของตนเอง ดังนั้นไฟล์ทั้งสองจึงไม่สามารถใช้งานกับซอฟต์แวร์ของเดิมได้อีกต่อไป ในทางกลับกันโปรแกรมป้ายประกาศก็ไม่สามารถใช้กับแฟ้มภาพของต่างประเทศดังกล่าวได้เช่นกัน

เมื่อได้เลือกแฟ้มภาพตาม 4 ขั้นตอนดังกล่าวแล้วก็จะมาถึงการเลือกภาพที่ต้องการ โดยโปรแกรมจะแสดงข้อความให้ทราบว่าเป็นภาพลำดับที่เท่าใดในจำนวนทั้งหมดกี่ภาพ พร้อมกับแสดงให้ทราบว่าแป้นที่ใช้คือแป้นลูกศรทั้งสี่และแป้น [Enter] ให้ท่านกดแป้นลูกศร ซึ่งภาพบนจอจะเปลี่ยนไปดังนี้

กดแป้นลูกศรขวาจะเลื่อนลำดับภาพให้มากขึ้น 1 ลำดับ
กดแป้นลูกศรซ้ายจะเลื่อนลำดับภาพให้น้อยลง 1 ลำดับ
กดแป้นลูกศรขึ้นจะเลื่อนลำดับภาพให้ลดลง 10 ลำดับ
กดแป้นลูกศรลงจะเลื่อนลำดับภาพให้มากขึ้น 10 ลำดับ
กดแป้น [Enter] หมายถึงการเลือกภาพที่แสดงในขณะนั้น

ขนาดภาพที่เป็นบนจอภาพคือขนาดที่เลือกไว้ ตัวอย่างของจอภาพในการเลือกภาพที่จะมาเขียนประกอบกับข้อความในป้ายประกาศคือรูปที่ 2.10

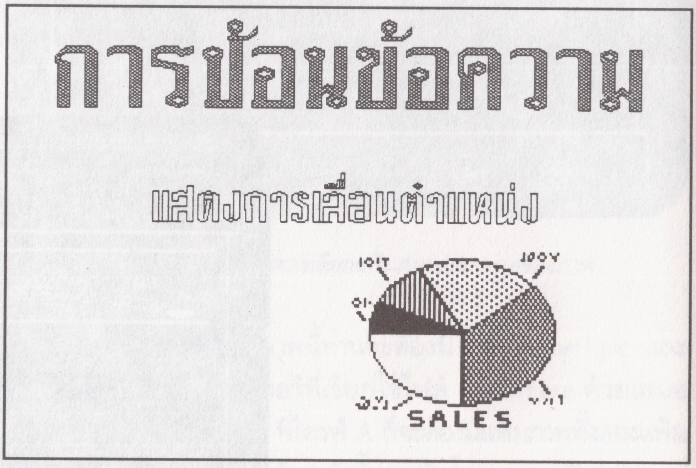


รูปที่ 2.10 *ส่วนของการเลือกลำดับที่ของภาพในแฟ้มภาพ*

หลังจากได้ภาพตามที่ต้องการแล้วก็จะมาถึงการเลือกตำแหน่งของการลงภาพ ซึ่งจะมีกรอบสี่เหลี่ยมเท่าขนาดจริงของภาพที่จะนำมาลงให้ผู้เขียนป้ายประกาศเลือกตำแหน่งของการลงภาพโดยใช้แป้นลูกศรทั้งสี่เลื่อนกรอบสี่เหลี่ยมนี้ไปมาบนป้ายประกาศที่กำลังเขียน โดยภาพที่ลงนี้สามารถลงข้ามรอยต่อของจอภาพจากแผ่นหนึ่งไปยังอีกแผ่นหนึ่งได้ (โดยจะได้ภาพเมื่อพิมพ์ลงกระดาษเป็นภาพเดียวกันด้วย)

เมื่อต้องการดูว่าภาพจะลง ณ ตำแหน่งที่เลือกแล้วเหมาะสมเพียงใดให้กดแป้น [F10] ก่อน จนได้ภาพที่มีความพอใจก็ให้กดแป้น [Enter]

ตัวอย่างการลงภาพประกอบข้อความจะเป็นดังรูปที่ 2.11



รูปที่ 2.11 *แสดงผลการลงภาพในแแผ่นป้ายประกาศ*

การแก้ไขข้อความและภาพ 3

เมื่อเราได้เขียนป้ายประกาศที่ประกอบด้วยข้อความและภาพไปเรียบร้อยแล้ว เมื่อทำการดูผลบนจอภาพของป้ายประกาศทั้งแผ่นแล้วพบว่า มีบางส่วนที่ยังไม่ถูกใจก็สามารถที่จะแก้ไขหรือเปลี่ยนแปลงในแต่ละส่วนให้เป็นไปตามที่ต้องการได้ เช่น การเปลี่ยนจากข้อความแบบปกติไปเป็นข้อความที่มีความโค้งหรือแบบอื่น ๆ การเปลี่ยนตำแหน่งการแสดงผล การเปลี่ยนแปลงข้อความที่ผิด เพิ่มเติมหรือตัดทิ้งในบางส่วน

การแก้ไขหรือเปลี่ยนแปลงป้ายประกาศของเดิมนี้สามารถทำกับป้ายประกาศที่ได้เก็บลงแผ่นบันทึกไปแล้วก็ได้ โดยการอ่านป้ายประกาศนั้นมาแก้ไขให้ตรงกับความต้องการ โดยสามารถกระทำได้ 2 ลักษณะคือ

1. แก้ไขหรือเปลี่ยนแปลงข้อความ
2. แก้ไขหรือเปลี่ยนแปลงรูปภาพ

โดยมีรายละเอียดดังนี้

การแก้ไขข้อความ การแก้ไขข้อความกระทำได้ดังนี้

❖ เลื่อนเมนูหลักไปยังส่วนการเขียนข้อความแล้วกดแป้น [Enter] โปรแกรมจะเข้าสู่เมนูย่อยของส่วนที่เกี่ยวกับการเขียนข้อความและรูปภาพ (ดังรูปที่ 2.1)

❖ กดแป้นลูกศรขึ้นลงเพื่อเลื่อนให้แถบสว่างของเมนูย่อยมาอยู่ที่รายการของการแก้ไขข้อความ แล้วกดแป้น [Enter]

❖ ต้องมีข้อความในป้ายประกาศนั้น ๆ อย่างน้อย 1 ข้อความ ถ้ายังไม่มีข้อความใด ๆ โปรแกรมจะคืนกลับไปยังเมนูหลักโดยไม่มีข้อความใด ๆ แจ้งให้ทราบ แต่ถ้ามีข้อความของป้ายประกาศอยู่แล้วก็จะเป็นการเลือกข้อความที่จะแก้ไข โดยจะมีการทำงานดังนี้

1. แสดงข้อความลำดับที่ 1 ในลักษณะกลับขาวเป็นดำ และมีข้อความบอกว่ามีข้อความทั้งหมดเท่าใด โดยจะมีข้อความแสดงถึงแป้นที่ใช้ในการเลือกด้วย

2. ให้กดแป้นลูกศรเพื่อเลือกข้อความที่ต้องการ โดยโปรแกรมจะแสดงข้อความเรียงลำดับตามแป้นที่กด โดยจะแสดงข้อความแบบกลับขาวดำที่ด้านบนของจอภาพ

3. เมื่อต้องการเลือกข้อความใดก็ให้กดแป้น [Enter] เมื่อข้อความนั้นปรากฏบนจอภาพ

❖ เมื่อได้เลือกข้อความแล้วจะมีเมนูย่อยซ้อนขึ้นมาให้เลือกอีกเมนูหนึ่ง ซึ่งจะมีรายการให้เลือกจำนวน 5 รายการคือ

1. แก้ไขข้อความ หมายถึงการแก้ไขในส่วนของข้อความ เช่น เปลี่ยนข้อความ เพิ่มเติมข้อความ ตัดข้อความบางส่วนออก เปลี่ยนฟอนต์ตัวหนังสือ เปลี่ยนแปลงตัวขยาย หรือตัวสูง 2 เท่า ฯลฯ กล่าวคือเป็นการเปลี่ยนแปลงในส่วนของข้อความที่อ่านได้หรือเปลี่ยนแปลงรหัสควบคุมการแสดงผล

2. กำหนดขนาดใหม่ หมายถึงการเลือกขนาดของข้อความใหม่

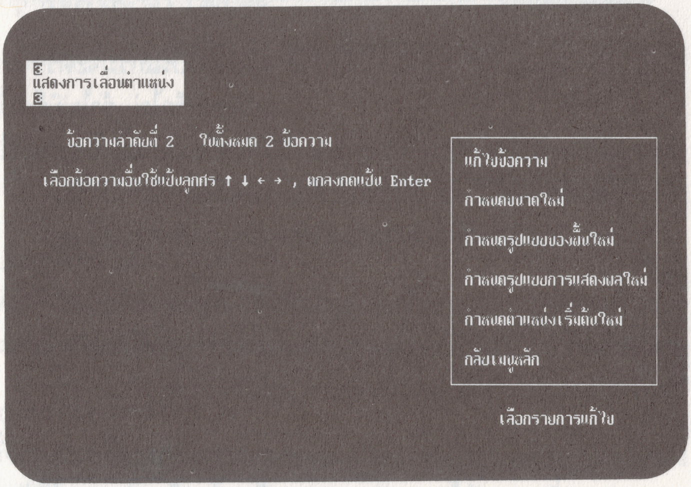
3. กำหนดรูปแบบของพื้นใหม่ หมายถึงการเปลี่ยนแปลงส่วนของพื้นของตัวอักขระในข้อความนั้น ๆ

4. กำหนดรูปแบบการแสดงผลใหม่ หมายถึงการเลือกแสดงข้อความในแบบต่าง ๆ เช่น โค้งขึ้น แบบคลื่น แบบเฉียง ฯลฯ

5. กำหนดตำแหน่งเริ่มต้นใหม่ หมายถึงการเปลี่ยนจุดแสดงข้อความบนป้ายประกาศจากจุดเดิมไปยังจุดใหม่เพื่อให้ได้ป้ายประกาศที่ดีขึ้น

โดยลักษณะของการเลือกรายการ ยังคงเป็นการใช้แป้นลูกศรเลื่อนแถบสว่างมาให้ตรงกับรายการที่เลือกแล้วกดแป้น [Enter] เช่นเดิม

ลักษณะที่ปรากฏบนจอภาพจะเป็นดังรูปที่ 3.1 ซึ่งกำลังเลือกแก้ไขข้อความว่า แสดงการเลื่อนตำแหน่ง ซึ่งเป็นข้อความลำดับที่ 2



รูปที่ 3.1 *เมนูย่อยในการแก้ไขข้อความของป้ายประกาศ*

❖ การแก้ไขข้อความ เมื่อเลือกรายการย่อยนี้ ขั้นตอนการทำงานคือ

1. แสดงข้อความที่จะแก้ไขให้ทราบ

2. มีเมนูย่อยให้เลือกว่าจะแก้ไขแบบค้นหาหรือแก้ไขทั้งบรรทัด ให้เลือกการแก้ไขโดยการใช้แป้นลูกศรและแป้น [Enter]

3. ถ้าเลือกการแก้ไขแบบค้นหา จะแสดงให้ทราบว่าอยู่ในส่วนการแก้ไขแบบค้นหา พร้อมกับแสดงข้อความที่แก้ไขในบรรทัดถัดมา บรรทัดล่างสุดจะแสดงรายการที่จะต้องป้อนข้อความ คือ ข้อความ _____ แก้ไขเป็น_____โดยจะมีเคอร์เซอร์กะพริบหลังคำว่าข้อความ ให้ป้อนข้อความที่ต้องการแก้ไขแล้วกดแป้น [Enter] เคอร์เซอร์ก็จะมากะพริบหลังคำว่า *แก้ไข เป็น ให้ป้อนข้อความใหม่*

โดยการป้อนข้อความนี้สามารถใช้รหัสควบคุมต่าง ๆ ได้เหมือนกับการป้อนข้อความในช่วงแรกทุกประการ

จากนั้นโปรแกรมจะตรวจว่าข้อความนั้น มีคำหรือข้อความที่ต้องการเปลี่ยนหรือไม่ ถ้าไม่มีก็จะไม่มีข้อความใด ๆ แจ้งให้ทราบ แต่ถ้ามีจะมีข้อความถามย้ำในการแก้ไขว่า โปรดยืนยันการแก้ไข (Y/N) ให้เลือกตามต้องการ

หากข้อความที่แก้ไขนี้มีรูปแบบการแสดงผลแบบโค้ง โปรแกรมจะทำการคำนวณส่วนของความโค้งให้ใหม่ทุกครั้งที่มีการแก้ไขข้อความ ทั้งนี้เพื่อให้ได้ความโค้งที่มีความสมมาตรกันทุกครั้ง

4. ถ้าเลือกการแก้ไขทั้งบรรทัด โปรแกรมจะเข้าสู่การป้อนข้อความทั้งบรรทัดแบบเดียวกับรูปที่ 2.2 เพียงแต่ข้อความบรรทัดแรกจะแสดงข้อความเดิมแทน และให้ป้อนข้อความใหม่แบบเดียวกับการป้อนข้อความในครั้งแรก

❖ การเลือกตามรายการในข้อ 5.3 จะได้ข้อความดังในรูปที่ 3.2

รูปที่ 3.2 *การแก้ไขข้อความแบบค้นหา*

❖ การกำหนดขนาดใหม่ เมื่อเลือกเปลี่ยนแปลงขนาดของข้อความก็จะมีวิธีการเลือกขนาดแบบเดียวกับรูปที่ 2.4 ให้เลือกขนาดโดยการกดแป้นลูกศรจนได้ขนาดที่ต้องการแล้วให้กดแป้น [Enter]

❖ การกำหนดรูปแบบของพื้นใหม่ จะเป็นการเลือกรูปแบบพื้นของตัวอักษรในข้อความนั้นในลักษณะเดียวกับรูปที่ 2.5 ให้เลือกโดยการกดแป้นลูกศรจนได้รูปแบบพื้นที่ต้องการแล้วให้กดแป้น[Enter]

❖ การกำหนดรูปแบบการแสดงผลใหม่ จะเป็นการเลือกรูปแบบของข้อความทั้งหมด โดยเลือกรูปแบบข้อความได้ตามรูปที่ 2.6 และ 2.7 โดยการใช้แป้นลูกศรจนปรากฏข้อความตามต้องการแล้วกดแป้น [Enter]

❖ การกำหนดตำแหน่งเริ่มต้นใหม่ เป็นการเลื่อนข้อความให้ไปแสดงที่จุดตั้งต้นใหม่ ซึ่งจะสามารถใช้แป้นควบคุมได้เหมือนกับตอนป้อนข้อความ คือ

แป้นลูกศรเลื่อนกรอบสี่เหลี่ยม (ตัวแทนของข้อความ) ไปยังตำแหน่งที่ต้องการ

แป้น [F8] เลื่อนเปลี่ยนอัตราการเคลื่อนย้ายข้อความเร็วหรือช้า
แป้น [F10] เพื่อดูผลของการลงข้อความ ณ ที่กรอบสี่เหลี่ยม
แป้น [Enter] สำหรับเลือกตำแหน่งของการลงข้อความ

เมื่อเสร็จสิ้นรายการทำงานทั้งหมดข้างต้น จะมีเมนูย่อยให้เลือกอีก 2 รายการคือ กลับเมนูหลักและแก้ไขข้อความต่อให้เลือกตามความต้องการ ถ้าเป็นการเลือกแก้ไขข้อความต่อก็จะย้อนกลับไปยังลำดับที่ 1 ใหม่ เพื่อเลือกการทำงานตามที่กล่าวมาทั้งหมดอีกครั้งหนึ่ง

หากเลือกกลับเมนูหลักก็จะเป็นการสิ้นสุดการแก้ไขข้อความ

การแก้ไขภาพประกอบ

เมื่อต้องการจะแก้ไขภาพที่ประกอบกับป้ายประกาศนั้น ก็ให้กลับเข้าสู่เมนูหลักก่อน แล้วดำเนินการตามขั้นตอนต่อไปนี้

❖ เลื่อนเมนูหลักไปยังส่วนการเขียนข้อความแล้วกดแป้น [Enter] จะเข้าสู่เมนูย่อยของส่วนที่เกี่ยวกับการเขียนข้อความและรูปภาพ (ดังรูปที่ 2.1)

❖ กดแป้นลูกศรขึ้นลงเพื่อเลื่อนให้แถบสว่างของเมนูย่อยมาอยู่ที่รายการของการแก้ไขภาพ แล้วกดแป้น [Enter]

❖ ให้เลือกการทำงานตามเมนูย่อย 2 รายการคือ ลงภาพใหม่และแก้ไขภาพเดิม ให้เลือกการแก้ไขภาพเดิมแล้วกดแป้น [Enter]

❖ จากนี้จะมาถึงส่วนของการเลือกภาพที่จะแก้ไข ซึ่งจะมีภาพเดิมปรากฏบนจอภาพเพื่อให้เลือกภาพที่ต้องการมาแก้ไข ดังรูปที่ 3.3

รูปที่ 3.3 *การเลือกภาพเดิมเพื่อแก้ไขขนาดหรือใช้ภาพใหม่*

โดยบนจอภาพจะแจ้งให้ทราบว่าเป็นภาพที่เท่าใด และมาจากแฟ้มภาพใด ถ้าต้องการเลื่อนไปเลือกภาพอื่นก็ให้กดแป้นใดแป้นหนึ่ง เมื่อต้องการแก้ไขภาพที่อยู่บนจอภาพก็ให้กดแป้น [Enter]

หากต้องการเปลี่ยนเพียงขนาดของภาพเท่านั้น ก็ให้จำด้วยว่าภาพที่เลือกนั้นเป็นของแฟ้มภาพหมายเลขใดด้วย

❖ ส่วนที่ตามมาคือการเลือกแฟ้มภาพและเลือกขนาด ดังรูปที่ 3.4

รูปที่ 3.4 *ส่วนการเลือกแฟ้มภาพและขนาดของจอภาพที่จะแก้ไข*

หากท่านต้องการแก้ไขขนาดเพียงอย่างเดียวให้เลือกแฟ้มหมายเลขเดิมและเปลี่ยนขนาดใหม่ก็จะได้ภาพเดิมมาแสดงเป็นภาพแรก

❖ ให้เลือกภาพที่ต้องการ โดยใช้แป้นลูกศรและแป้น [Enter]

❖ เลือกตำแหน่งที่จะแสดงภาพ โดยตำแหน่งจะเริ่มต้น ณ ตำแหน่งเดิมของภาพที่นำมาแก้ไข ให้เลือกตำแหน่งของภาพเหมือนกับตอนที่ลงภาพ โดยใช้แป้นลูกศร แป้น [F8], [F10] และแป้น [Enter]

เมื่อการทำงานสิ้นสุดก็จะเป็นการจบส่วนของการแก้ไขภาพเดิม

การลบข้อความเดิม

หากเราได้เขียนป้ายประกาศหนึ่งป้ายลงในระบบการทำงานของโปรแกรมแล้ว ถ้ามีการป้อนข้อความหรือรูปภาพก็จะเป็นการกระทำกับป้าย-ประกาศแผ่นเดิม การเก็บป้ายประกาศนั้นลงในแผ่นบันทึกไม่ใช่เป็นการสิ้นสุดการเขียนป้ายประกาศนั้น ซึ่งทำให้การเขียนป้ายประกาศแผ่นใหม่จะต้องทำตามขั้นตอนต่อไปนี้

- ❖ เลื่อนเมนูหลักไปยังส่วนการเขียนข้อความแล้วกดแป้น [Enter] จะเข้าสู่เมนูย่อยของส่วนที่เกี่ยวกับการเขียนข้อความและรูปภาพ (ดังรูปที่ 2.1)
- ❖ กดแป้นลูกศรขึ้นลงเพื่อเลื่อนให้แถบสว่างของเมนูย่อยมาอยู่ที่รายการของการลบข้อความเดิม แล้วกดแป้น [Enter]
- ❖ จะมีเมนูย่อยแสดงรายการให้เลือก 2 รายการคือ ลบข้อความเดิม และกลับเมนูย่อย ให้เลือกรายการตามต้องการ
- ❖ เมื่อเลือกลบข้อความเดิม โปรแกรมจะจัดการกับตัวแปรที่ใช้ให้มาอยู่ในสภาวะเตรียมพร้อมเพื่อเขียนป้ายประกาศแผ่นใหม่
- ❖ เมื่อจบการทำงานจากลำดับที่ 3 หรือ 4 จะกลับมายังเมนูย่อยของการป้อนข้อความ ซึ่งจะเลือกรายการป้อนข้อความต่อเลยก็ได้
- ❖ จบส่วนการลบข้อความเดิมเพียงแค่นี้

การเลือกรูปแบบพื้น

เพื่อให้ป้ายประกาศมีหน้าตาที่แปลกขึ้น เราสามารถที่จะเปลี่ยนพื้นกระดาษจากที่ให้โล่ง ๆ ไว้มาเป็นการกำหนดรูปแบบของพื้นได้ ซึ่งจะมีผลให้ข้อความและรูปภาพทั้งหมดไปเขียนทับบนพื้นที่ที่เราได้กำหนดรูปแบบของพื้นไว้แล้ว

การทำแบบนี้เป็นคนละส่วนกับการกำหนดรูปแบบพื้นของตัวอักขระในเรื่องของการป้อนข้อความหรือแก้ไขข้อความ โดยมีขั้นตอนการทำงานดังนี้

❖ เลื่อนเมนูหลักไปยังส่วนการเขียนข้อความแล้วกดแป้น [Enter] ก็จะเป็นการเข้าสู่เมนูย่อยของส่วนที่เกี่ยวกับการเขียนข้อความและรูปภาพ (ดังรูปที่ 2.1)

❖ กดแป้นลูกศรขึ้นลงเพื่อเลื่อนให้แถบสว่างของเมนูย่อยมาอยู่ที่รายการของการเลือกรูปแบบพื้น แล้วกดแป้น [Enter]

❖ โปรแกรมจะมีรูปแบบพื้นปรากฏในกรอบสี่เหลี่ยมให้เลือก ดังรูปที่ 3.5

รูปที่ 3.5 *การเลือกรูปแบบพื้นของป้ายประกาศทั้งแผ่น*

ให้เลือกรูปแบบพื้นที่ต้องการด้วยการกดแป้นลูกศร และเลือกรูปแบบพื้นนั้น ๆ ด้วยการกดแป้น [Enter] การเลือกรูปแบบพื้นนี้ จะปรากฏผลบนจอภาพในกรณีที่ดูป้ายประกาศและพิมพ์ลงกระดาษ

ตัวอย่างของป้ายประกาศที่กำหนดพื้นกระดาษให้เปลี่ยนไปเป็นดังรูปที่ 3.6

รูปที่ 3.6 *ภาพตัวอย่างการกำหนดพื้นของป้ายประกาศ*

4
การพิมพ์ป้ายประกาศ

หลังจากที่ได้ป้อนข้อความและลงภาพประกอบในป้ายประกาศเสร็จแล้ว ก็จะถึงการพิมพ์ป้ายประกาศนั้น ๆ ลงกระดาษเพื่อนำไปใช้ประกาศต่อไป ซึ่งมีสิ่งที่ควรรู้ก่อนที่จะพิมพ์ป้ายประกาศดังนี้

❖ การพิมพ์ป้ายประกาศนี้จะเป็นการสำเนาจอภาพทั้งจอลงไปยังกระดาษ ดังนั้นการทำงานจึงมีลักษณะของโปรแกรมประเภท WYSIWYG นั่นคือส่วนที่ปรากฏบนจอภาพเป็นอย่างไรก็จะได้ป้ายประกาศในลักษณะเดียวกัน

ดังนั้นท่านจะต้องดูผลของป้ายประกาศนั้น ๆ ก่อนที่จะพิมพ์ลงบนกระดาษทุกครั้ง

❖ เครื่องพิมพ์ที่ใช้พิมพ์ต้องเป็นเครื่องพิมพ์แบบหัวเข็มพิมพ์ ใช้ได้ทั้งแบบ 9 หัวเข็มและ 24 หัวเข็ม แต่จะได้ภาพมีความยาวต่างกันเล็กน้อย สำหรับเครื่องพิมพ์แบบเลเซอร์นั้น ในขณะนี้โปรแกรม Thai Banner ยังไม่สามารถใช้งานได้ คงต้องรอโอกาสให้ผู้เขียนมีเครื่องพิมพ์เลเซอร์ใช้ก่อนนะ...

❖ ภาพประกอบในป้ายประกาศนั้นจะถูกกำหนดให้มีตัวชี้ไว้ที่ข้อความใดข้อความหนึ่งให้โปรแกรมทราบว่ามีภาพต่อจากข้อความนั้น ๆ และในแต่ละภาพจะมีตัวชี้ให้ทราบอีกว่ามีภาพอื่น ๆ ที่จะแสดงต่ออีกหรือไม่ ดังนั้นการแสดงภาพจึงจะมีความสมบูรณ์เสมอไม่ว่าตำแหน่งแสดงของภาพจะอยู่ ณ ที่ใด

❖ ข้อความที่เป็นภาษาไทยและกำหนดขนาดใหญ่โตมาก ๆ อาจจะแสดงผลได้ไม่ครบถ้วนสมบูรณ์ เช่น อาจมีส่วนของสระและวรรณยุกต์ขาดหายไปได้ จึงขอให้สังเกตป้ายประกาศให้ดีก่อนการพิมพ์ลงกระดาษด้วย

การดูป้ายประกาศบนจอภาพ

ทุก ๆ ครั้งที่เราไม่แน่ใจว่าได้รูปแบบของป้ายประกาศตรงกับความต้องการหรือไม่ ก็สามารถดูป้ายประกาศนั้น ๆ ได้โดยการทำตามขั้นตอนต่อไปนี้

❖ เข้ามาสู่เมนูหลักแล้วเลื่อนแถบสว่างมาอยู่ที่รูปเครื่องคอมพิวเตอร์หรือเมนูหลักรายการเลือกที่ 2 แล้วให้กดแป้น [Enter]

❖ ถ้ายังไม่มีการเขียนข้อความใด ๆ จะมีข้อความแจ้งให้ทราบพร้อมกับมีเสียงดังอยู่ระยะหนึ่งแล้วจึงกลับคืนมายังเมนูหลัก

❖ ถ้ามีการป้อนข้อความหรือภาพแล้วก็จะแสดงป้ายประกาศเรียงลำดับจากแผ่นที่ 1 ไปจนถึงแผ่นสุดท้าย โดยจะมีข้อความแจ้งให้ทราบดังนี้

1. ทุก ๆ แผ่นจะมีข้อความแจ้งที่มุมบนซ้ายให้ทราบว่าเป็นแผ่นที่เท่าใด และให้ข้อความว่า กดแป้นใด ๆ (Esc==>กลับ) ซึ่งจะเป็นข้อความแบบกะพริบ

2. สำหรับกรณีที่ไม่ใช่แผ่นสุดท้ายของป้ายประกาศจะมีข้อความแจ้งที่มุมบนขวาว่า ยังมีต่อ==>

❖ การใช้แป้นในการดูป้ายประกาศทำได้ดังนี้

1. กดแป้นใดก็ได้จะเป็นการดูป้ายประกาศในแผ่นถัดไป
2. กดแป้น [Esc] ณ แผ่นใดก็ได้จะเป็นการกลับไปยังเมนูหลัก
3. ถ้าเป็นแผ่นสุดท้ายจะกดแป้นใดก็จะกลับคืนเมนูหลักเช่นกัน

ให้สังเกตผลของป้ายประกาศบนจอภาพด้วยว่าให้ผลตามต้องการหรือไม่ หากเห็นว่ายังไม่สมบูรณ์ก็ให้กลับไปแก้ไขข้อความและภาพตามบทที่ 3 อีกครั้งหนึ่ง หากได้ป้ายประกาศตามความต้องการแล้วก็สามารถพิมพ์ลงกระดาษได้

เมนูส่วนการพิมพ์

เมื่อได้ป้ายประกาศตามความต้องการแล้ว หากประสงค์จะพิมพ์ป้ายประกาศนั้น ๆ ลงกระดาษก็ให้ทำตามขั้นตอนต่อไปนี้

❖ เข้ามาสู่เมนูหลักแล้วเลื่อนแถบสว่างมาอยู่ที่รูปเครื่องพิมพ์หรือเมนูหลักของรายการเลือกที่ 3 แล้วให้กดแป้น [Enter]

❖ ถ้าไม่มีการเขียนข้อความใด ๆ จะมีข้อความแจ้งให้ทราบแล้วกลับสู่เมนูหลัก

❖ ถ้าได้เขียนป้ายประกาศแล้วจะตรงไปยังเมนูย่อยการพิมพ์ ดังรูปที่ 4.1

รูปที่ 4.1 *เมนูย่อยแสดงรายการพิมพ์ลงกระดาษ*

โดยมีรายการให้เลือกดังนี้

1. กระดาษกว้าง 8 นิ้ว
2. กระดาษกว้าง 5 นิ้ว
3. กระดาษกว้าง 4 นิ้ว
4. กระดาษกว้าง 3 นิ้ว
5. พิมพ์ปกติทีละ 1 หน้า
6. กลับเมนูหลัก

สำหรับการพิมพ์ป้ายประกาศนี้ในการทำงานผู้เขียนยึดเอาเครื่องพิมพ์ชนิดแคร่สั้นเป็นหลัก เพราะจะสัมพันธ์กับจอภาพที่จะลอกลงกระดาษได้ดีที่สุด แต่ก็ยังเอื้ออำนวยต่อการพิมพ์ในเครื่องพิมพ์ชนิดแคร่ยาว โดยการให้สามารถกำหนดสดมภ์กั้นซ้ายได้อีก ซึ่งจะช่วยให้พิมพ์ป้ายประกาศ 2 ครั้งในกระดาษที่กว้างกว่า 8 นิ้วได้

การพิมพ์ป้ายประกาศนี้สามารถทำได้ 2 ลักษณะคือ

- ❖ รายการที่ 3-4 จะเป็นการพิมพ์ตามขวาง (หมุนจอภาพไป 90° แล้วลอกลงกระดาษ) กรณีนี้จะต้องใช้กับกระดาษต่อเนื่องเท่านั้น ขอให้ตรวจกระดาษต่อเนื่องที่ใช้ว่ามีรอยปรุของขอบกระดาษแข็งแรงพอที่จะดึงกระดาษให้ขยับไปได้โดยไม่ติดขัด หากมีการติดขัดของกระดาษจะทำให้การพิมพ์ป้ายประกาศนั้น ๆ มีปัญหาได้

- ❖ รายการที่ 5 จะเป็นการลอกจอภาพในลักษณะที่เห็นอย่างไรบนจอภาพก็จะได้ป้ายประกาศบนกระดาษในแนวเดียวกัน การพิมพ์แบบนี้จะต้องคำนึงถึงเรื่องเรื่องหนึ่งคือ ป้ายประกาศทุก ๆ แผ่นบนจอภาพจะต้องมีความสมบูรณ์ในแผ่นนั้น ๆ จะต้องไม่มีข้อความหรือภาพเหลื่อมซ้อนล้ำเข้าไปในแผ่นอื่น ๆ

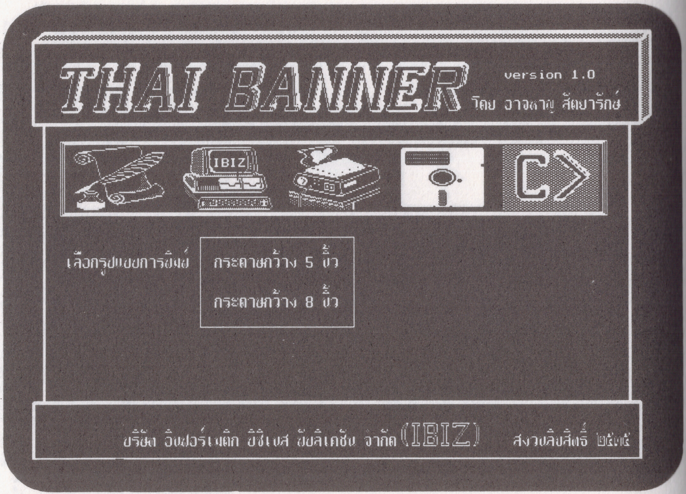
การพิมพ์ก็จะทำได้ 2 กรณี คือ

- ❖ กรณีแรก ถ้าป้ายประกาศที่เขียนนั้นมีไม่เกิน 2 จอภาพ จะสามารถพิมพ์ลงบนกระดาษแผ่นเดียวหรือกระดาษไขสำหรับนำไปโรเนียวได้ จึงสามารถทำปกรายงานหรือปกหนังสือได้

- ❖ กรณีที่สอง ถ้าป้ายประกาศมีมากกว่า 2 จอภาพ ต้องใช้กระดาษต่อเนื่องเท่านั้นเพราะว่าไม่มีการกดแป้นเพื่อรอการเปลี่ยนกระดาษ ซึ่งจะเป็นการพิมพ์อย่างต่อเนื่องจากต้นไปจนจบป้ายประกาศ

โดยเมื่อเลือกการพิมพ์แบบนี้จะมีเมนูย่อยให้เลือกความกว้างของกระดาษดังรูปที่ 4.2 ซึ่งจะมี 2 รายการ พร้อมกับให้กำหนดตำแหน่งกั้นซ้ายในการพิมพ์ด้วย

หากมาผิดทางให้เลื่อนแถบสว่างไปที่ กลับเมนูหลัก แล้วกด [Enter] จะคืนไปยังเมนูหลักเพื่อทำงานด้านอื่น ๆ ต่อไป



รูปที่ 4.2 *เมนูย่อยเลือกความกว้างของกระดาษในกรณีเลือกพิมพ์ในแนวปกติ*

การพิมพ์ปกหนังสือ

เนื่องจากโปรแกรม Thai Banner ได้ออกแบบให้สามารถใช้งานได้มากแบบอยู่แล้ว เช่น สามารถนำมาพิมพ์เป็นปกหนังสือหรือปกรายงานได้ เราจึงมาดูแนวคิดในการสร้างปกรายงานสัก 1 ตัวอย่าง

สมมติว่ามีความต้องการจะทำปกรายงาน โดยข้อความทั้งหมดมีดังนี้

รายงานวิชาฟิสิกส์

เรื่อง การสื่อสารในประเทศไทย

โดย อาจหาญ สัตยารักษ์

รวม 3 ข้อความ

เราจะทำงานตามลำดับดังนี้

- ❖ ป้อนข้อความครั้งละ 1 ข้อความ เลือกขนาด รูปแบบพื้น รูปแบบการแสดงผล และตำแหน่งของข้อความบนจอภาพ โดยข้อความ

ทั้งหมดจะต้องแสดงจบในหนึ่งจอภาพเท่านั้น ทำดังนี้จนกระทั่งครบทั้ง 3 ข้อความ

ตามตัวอย่างจะกำหนดข้อความดังนี้

รายงานวิชาฟิสิกส์ กด [F2] เพื่อแสดงข้อความด้วยฟอนต์ที่สอง ให้กด [^]-[S] เพื่อเป็นข้อความแบบตัวนูน เลือกรูปแบบพื้นแบบว่าง (คือไม่มีพื้นของตัวอักขระ) และแสดงผลแบบนูนสองข้าง

เรื่อง การสื่อสารในประเทศไทย กด [F4] เพื่อแสดงข้อความด้วยฟอนต์ที่สี่ กำหนดรูปแบบพื้นของอักขระ และแสดงข้อความแบบบนเว้าล่างตรง เพื่อให้ข้อความที่สองนี้เข้ากันได้กับข้อความที่หนึ่ง

โดย อาจหาญ สัตยารักษ์ กด [F3] เพื่อแสดงข้อความด้วยฟอนต์ที่สาม แสดงข้อความแบบปกติด้วยพื้นว่าง

การจัดการกับจอภาพนั้น หากต้องการพิมพ์ให้เหมาะสมกับหน้ากระดาษควรจัดปกรายงานนี้ให้อยู่ใน 2 จอภาพ โดยจอภาพแรก (แผ่นที่ 1) จะเป็นส่วนบนของปกรายงาน และจอภาพที่สอง (แผ่นที่ 2) จะเป็นส่วนล่างของปกรายงาน

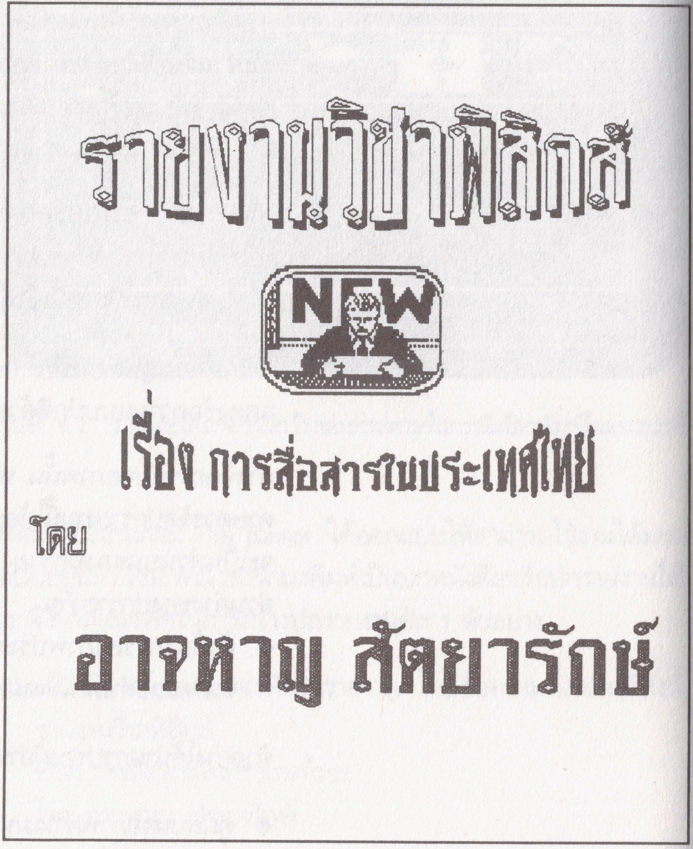
❖ ถ้าต้องการลงภาพประกอบก็ให้เลือกภาพตามต้องการ กำหนดขนาดและตำแหน่งตามที่เห็นเหมาะสม

ตัวอย่างได้นำเอารูปของผู้อ่านข่าวทางโทรทัศน์มาประกอบในป้ายประกาศ

❖ ดูผลบนจอภาพผ่านเมนูหลักรายการที่ 2 อีกครั้งหนึ่ง ทั้งนี้เพื่อให้เกิดความมั่นใจว่าได้ปกรายงานที่มีความเหมาะสมแล้ว

❖ เก็บป้ายประกาศที่เขียนแล้วนี้ในแฟ้มข้อมูลในชื่อ physics1.ban เพื่อที่อาจนำมาต่อเติมหรือแก้ไขในภายหลัง

❖ พิมพ์ปกรายงานนี้โดยการพิมพ์แบบกระดาษแผ่นเดียว ดังรูปที่ 4.3



รูปที่ 4.3 *ตัวอย่างการใช้ป้ายประกาศทำปกรายงาน*

การพิมพ์บนกระดาษที่กว้างกว่า 8 นิ้ว

ในกรณีที่เครื่องพิมพ์เป็นแคร่ยาวที่สามารถใช้ได้กับกระดาษต่อเนื่องที่มีความกว้างมากกว่า 8 นิ้ว เราสามารถพิมพ์ป้ายประกาศได้มากขึ้น โดยมีข้อปฏิบัติดังนี้

❖ เขียนป้ายประกาศแยกเป็น 2 ป้ายประกาศ โดยให้ข้อความของป้ายประกาศชุดแรกเป็นส่วนบนของป้ายประกาศ (เวลาพิมพ์เป็นส่วนขวามือของกระดาษต่อเนื่อง) เก็บป้ายประกาศนี้ไว้ในแฟ้มข้อมูลสมมติชื่อ test1.ban

ป้ายประกาศชุดที่สองเป็นส่วนล่างของป้ายประกาศ (เวลาพิมพ์จะเป็นส่วนซ้ายมือของกระดาษต่อเนื่อง) เก็บป้ายประกาศนี้ไว้ในแฟ้มข้อมูลอีกชื่อหนึ่ง สมมติชื่อ test2.ban

❖ เข้าสู่เมนูการพิมพ์ เลือกการพิมพ์แบบกระดาษกว้าง 8 นิ้ว (ขณะนี้มีป้ายประกาศจากแฟ้มชื่อ test2.ban อยู่ในหน่วยความจำ) ให้พิมพ์โดยเลือกตำแหน่งการพิมพ์ ณ จุดซ้ายสุดของกระดาษคือ 0 ทำการพิมพ์ป้ายประกาศจนเสร็จ ระหว่างการพิมพ์ให้สังเกตด้วยว่าข้อความของป้ายประกาศด้านขวาสุดอยู่ที่ตำแหน่งเท่าใดบนเครื่องพิมพ์ เพื่อจะนำไปกำหนดตำแหน่งกั้นซ้ายให้กับการพิมพ์ครั้งที่ 2

❖ เลื่อนกระดาษแผ่นเดิมนี้ให้กลับมาที่จุดตั้งต้นใหม่ (หรืออาจนำออกมาใส่ใหม่โดยไม่เลื่อนกระดาษถอยหลังก็ได้) จัดตั้งกระดาษให้ได้ตำแหน่งที่คิดว่าเหมาะสมกับการพิมพ์ป้ายประกาศของแฟ้มชื่อ test1.ban

❖ อ่านแฟ้มข้อมูลป้ายประกาศ test1.ban เข้าสู่หน่วยความจำ

❖ เลือกการพิมพ์ในลักษณะเดิมกับการเข้าสู่เมนูการพิมพ์พร้อมกับกำหนดตำแหน่งกั้นซ้ายให้เท่ากับตำแหน่งขวาสุดของการพิมพ์ในลำดับที่ 2 (พิมพ์ครั้งแรก)

- พิมพ์ข้อความของป้ายประกาศ test1.ban ไปจนจบ จะได้ป้ายประกาศบนกระดาษที่มีความกว้างมากกว่า 8 นิ้วได้

นอกจากนี้ท่านอาจเลือกพิมพ์จากกระดาษกว้าง 5, 4 หรือ 3 นิ้วได้ในลักษณะเดียวกันนี้ เช่น เลือกการพิมพ์บนกระดาษกว้าง 4 นิ้วสองครั้งเพื่อให้ป้ายประกาศของแฟ้มข้อมูล test1.ban และ test2.ban เป็นป้ายประกาศเดียวกันบนกระดาษต่อเนื่องขนาดกว้าง 8 นิ้วได้

ทั้งนี้อยู่ที่ดุลยพินิจของท่านในการออกแบบป้ายประกาศและเลือกวิธีการพิมพ์ป้ายประกาศเป็นสำคัญ

การทดลองเขียนป้ายประกาศและทดลองพิมพ์หลาย ๆ ครั้ง จะทำให้ท่านมองเห็นแนวทางที่เหมาะสมในการพิมพ์ป้ายประกาศด้วยตนเอง ซึ่งเป็นเรื่องที่น่าสนุกเพราะป้ายประกาศที่เขียนไว้เพียงแฟ้มข้อมูลเดียวนั้นเราสามารถนำมาพิมพ์เป็นรูปแบบของป้ายประกาศได้มากมายหลายแบบที่ไม่ซ้ำกัน

การอ่านและเขียนแฟ้มข้อมูลบนแผ่นบันทึก 5

เมื่อได้เขียนป้ายประกาศแผ่นหนึ่งแผ่นใดเรียบร้อยแล้ว หากต้องการจะเก็บป้ายประกาศแผ่นนั้นลงแผ่นบันทึก เพื่อนำกลับมาใช้อีกในคราวต่อไปก็สามารถทำได้ ไม่ว่าจะเป็นการเก็บลงแผ่นบันทึกแม่เหล็กธรรมดาผ่านไดรฟ์ A หรือ B หรือการเก็บลงฮาร์ดดิสก์

แฟ้มข้อมูลของป้ายประกาศที่จะเก็บป้ายประกาศมีลักษณะดังนี้

❖ เก็บแยกป้ายประกาศหนึ่งป้ายต่อหนึ่งแฟ้มข้อมูล

❖ ส่วนแรกของแฟ้มข้อมูลจะเป็นการเก็บข้อความเพื่อให้ทราบว่าเป็นแฟ้มข้อมูลของป้ายประกาศ

❖ ในแต่ละแฟ้มจะประกอบด้วยข้อมูลดังนี้

1. หมายเลขของรูปพื้นของป้ายประกาศ
2. จำนวนข้อความทั้งหมดในป้ายประกาศ
3. ข้อมูลของข้อความทั้งหมด ซึ่งประกอบด้วยจุดเริ่มต้น ข้อความ ขนาด หมายเลขรูปแบบพื้นของข้อความ รูปแบบการแสดงผล และตัวชี้ว่ามีภาพที่จะแสดงต่อจากข้อความนั้น ๆ หรือไม่
4. จำนวนหน้าจอภาพสูงสุดของป้ายประกาศ (จำนวนแผ่น)
5. จำนวนภาพที่ประกอบเป็นป้ายประกาศ

6. เก็บข้อมูลเกี่ยวกับภาพ เช่น จุดเริ่มต้น หมายเลขแฟ้มภาพ หมายเลขลำดับที่ของภาพ และขนาดของภาพ

❖ แฟ้มข้อมูลจะมีชื่อสกุลเป็น .ban โดยโปรแกรมจะเป็นผู้กำหนดให้ ไม่ว่าจะเป็นการอ่านหรือการเขียนก็ตาม

❖ ถ้าใช้คำสั่ง type กับแฟ้มข้อมูลของป้ายประกาศทุกแฟ้มจะได้ข้อความ 'DATA of ThaiBanner By Arthan Sattayarak ปรากฏบนจอภาพเท่านั้น

❖ ห้ามใช้โปรแกรมพิมพ์เอกสารเรียกแฟ้มข้อมูลนี้เข้ามาแล้วเก็บข้อมูลลงแฟ้มใหม่ เพราะจะทำให้ข้อมูลทั้งหมดสูญหายไป

เมื่อเราต้องการจัดการเกี่ยวกับแฟ้มข้อมูล ก็ให้เลื่อนแถบสว่างในเมนูหลักให้มากะพริบที่ภาพของแผ่นดิสก์หรือรายการเลือกที่ 4 ซึ่งบนจอภาพจะปรากฏรายการให้เลือกดังรูปที่ 5.1

รูปที่ 5.1 *เมนูย่อยในกรณีการอ่าน-เขียนแฟ้มข้อมูล*

โดยจะมีเมนูย่อยให้เลือก 3 รายการ คือ

❖ เก็บข้อมูลลงแฟ้มข้อมูล

❖ อ่านข้อมูลจากแฟ้มข้อมูล

❖ เปลี่ยนสารบัญไฟล์

เมื่อเข้าสู่เมนูย่อยจริง ๆ จะมีทางเลือกที่ 4 คือกลับเมนูหลักเพิ่มอีกหนึ่งรายการ เพื่อให้ผู้ที่เปลี่ยนใจในการกระทำกับแฟ้มข้อมูลจะได้มีหนทางย้อนกลับไปยังเมนูหลักได้ โดยให้เลือกการทำงานตามรายการที่ต้องการ

การเปลี่ยนสารบัญไฟล์

ต่อไปเราจะมาศึกษาเรื่องการเปลี่ยนสารบัญไฟล์ (หรือเปลี่ยนไดรฟ์) กันก่อน หากสังเกตเมนูของป้ายประกาศที่ได้แสดงมาทั้งหมดให้ดี ผู้อ่านจะพบว่าที่รายการสุดท้ายจะแสดง C> ซึ่งหมายถึงว่าขณะนั้นโปรแกรมกำลังติดต่อกับฮาร์ดดิสก์ C อยู่ โดยอักษรนี้จะเปลี่ยนแปลงไปเมื่อเราได้เปลี่ยนสารบัญไฟล์ ดังในรูปที่ 5.2

รูปที่ 5.2 *ภาพแสดงให้เห็นชื่อไดรฟ์ที่กำลังติดต่ออยู่*

โดยในรูปที่ 5.2 ได้คัดลอกหน้าจอภาพมา 2 จอภาพ ซึ่งจะแสดงให้เห็นว่าเป็นการติดต่อกับไดรฟ์ A และไดรฟ์ B ซึ่งชื่อไดรฟ์ที่ติดต่อนี้จะหมายถึงการอ่านและเขียนแฟ้มข้อมูลของป้ายประกาศเท่านั้น ไม่ใช่ไดรฟ์ที่จะอ่านแฟ้มภาพที่ใช้ประกอบกับป้ายประกาศแต่อย่างใด

เมื่อเราต้องการเปลี่ยนสารบัญไฟล์ให้ทำตามขั้นตอนต่อไปนี้

❖ เลื่อนเมนูหลักมายังภาพแผ่นดิสก์แล้วกดแป้น [Enter] จะปรากฏเมนูย่อยให้เลือกการทำงาน

❖ เลื่อนแถบสว่างมาที่ข้อความว่า เปลี่ยนสารบัญไฟล์ แล้วกดแป้น [Enter]

❖ โปรแกรมจะแสดงข้อความให้ทราบว่ากำลังอยู่ในสารบัญเดิมใด และแสดงรูปแบบของการป้อนสารบัญใหม่ พร้อมแจ้งให้ทราบว่าป้อน EXIT จะกลับคืนไปยังเมนูหลัก

❖ ป้อนสารบัญที่ต้องการเปลี่ยนไป โดยต้องป้อนให้ตรงกับรูปแบบที่กำหนด

❖ ถ้าป้อนไม่ตรงกับรูปแบบที่กำหนดจะต้องกลับมาป้อนใหม่ โดยไม่มีข้อความใด ๆ แจ้งให้ทราบ

❖ ถ้าป้อนตรงกับรูปแบบที่กำหนด แต่โปรแกรมทำการเปลี่ยนสารบัญแล้วพบว่าไม่มีสารบัญที่ระบุจะมีข้อความแจ้งให้ทราบว่า ไม่มีสารบัญ-ไฟล์ แล้ววนกลับมาให้ป้อนสารบัญไฟล์ใหม่อีกครั้ง

❖ ถ้าป้อนตรงรูปแบบถูกต้องและค้นหาสารบัญพบก็จะเปลี่ยนสารบัญให้ โดยถ้าชื่อสารบัญชื่อแรกที่ป้อนคือ A, B ,C, D ฯลฯ ไม่ตรงกับชื่อเดิมก็จะมีการเปลี่ยนเมนูหลักรายการสุดท้ายให้ตรงกับไดรฟ์ที่กำลังติดต่อ

รูปที่ 5.3 เป็นตัวอย่างหนึ่งของจอภาพในขณะที่มีการเปลี่ยนสารบัญไฟล์

รูปที่ 5.3 *ตัวอย่างการเปลี่ยนสารบัญไฟล์*

การเขียนแฟ้มข้อมูลโปรแกรม Thai Banner

ป้ายประกาศที่เราเขียนเรียบร้อยแล้วหรือเขียนไปได้ส่วนหนึ่ง หากเกรงว่าจะมีปัญหาเรื่องไฟฟ้าดับ หรือได้รูปแบบของป้ายประกาศส่วนหนึ่งเป็นที่พอใจแล้วแต่ไม่แน่ใจว่าเมื่อเขียนป้ายประกาศต่อจะทำให้รูปแบบดีหรือไม่ ก็อาจเก็บป้ายประกาศนั้น ๆ ลงแผ่นบันทึกเสียก่อน โดยจะเก็บลงแผ่นบันทึกหรือฮาร์ดดิสก์ก็แล้วแต่ว่าเครื่องคอมพิวเตอร์ของท่านมีอุปกรณ์ประกอบอย่างไร หรือแล้วแต่จุดประสงค์ของท่าน

การเก็บข้อมูลของป้ายประกาศลงแผ่นบันทึกทำได้ดังนี้

❖ เลื่อนเมนูหลักมายังภาพแผ่นดิสก์แล้วกดแป้น [Enter] จะปรากฏเมนูย่อยให้เลือกการทำงาน

❖ เลื่อนแถบสว่างมาที่ข้อความว่า เก็บข้อมูลลงแฟ้มข้อมูล แล้วกด [Enter]

❖ โปรแกรมจะให้ป้อนชื่อไฟล์จำนวนไม่เกิน 8 อักขระ (ต้องตั้งชื่อตามข้อกำหนดของดอสด้วย) ในโปรแกรมสามารถป้อน : \ ได้ แต่จะถูกสั่งให้ป้อนใหม่ทั้งนี้เนื่องจากไดเรกทอรีนั้นโปรแกรมจะตรวจสอบให้เองอยู่แล้ว ดังนั้นแม้สองอักขระนี้จะป้อนได้ก็ไม่ต้องป้อน การป้อนชื่อไฟล์นี้ไม่ต้องใส่ชื่อสกุล เพราะโปรแกรมจะจัดการให้เองทั้งหมด

❖ เมื่อกดแป้น [Enter] หลังการป้อนชื่อไฟล์ จะมีการตรวจสอบว่ามีไฟล์ชื่อนี้ในสารบัญไฟล์ที่กำลังติดต่ออยู่หรือไม่ ถ้ามีจะมีข้อความเมนูย่อยปรากฏบนจอภาพ ให้เลือกการทำงานดังรูปที่ 5.4 จำนวน 3 รายการ คือ

1. ป้อนชื่อไฟล์ใหม่ จะกลับไปป้อนชื่อไฟล์ตามลำดับที่ 3 ใหม่
2. ต้องการเขียนทับ จะเขียนทับในไฟล์ที่ระบุ
3. กลับเมนูหลัก กลับไปเมนูหลักโดยไม่มีการเขียนไฟล์ใด ๆ

ให้เลือกการทำงานตามต้องการ

รูปที่ 5.4 *เมนูย่อยในกรณีที่ป้อนชื่อไฟล์ซ้ำกับที่มีอยู่แล้ว*

การอ่านแฟ้มข้อมูล
โปรแกรม Thai Banner

บางครั้งเราอาจจะต้องนำป้ายประกาศที่เขียนไว้เดิมมาพิมพ์ลงกระดาษใหม่ หรืออาจต้องการนำป้ายประกาศแผ่นเก่ามาแก้ไขเพิ่มเติมเพื่อช่วยร่นระยะเวลาในการเขียนป้ายประกาศให้สั้นเข้า แฟ้มข้อมูลของป้าย-ประกาศเดิมจึงจะถูกอ่านเข้ามาใช้งานใหม่

ขั้นตอนในการอ่านแฟ้มข้อมูลของป้ายประกาศเดิมมีดังนี้

❖ เลื่อนเมนูหลักมายังภาพแผ่นดิสก์แล้วกดแป้น [Enter] จะปรากฏเมนูย่อยให้เลือกการทำงาน

❖ เลื่อนแถบสว่างมาที่ข้อความว่า อ่านข้อมูลจากแฟ้มข้อมูล แล้วกด [Enter]

❖ โปรแกรมจะอ่านแฟ้มข้อมูลที่มีในสารบัญที่กำลังติดต่ออยู่นั้นเข้ามายังหน่วยความจำพร้อมกับแสดงชื่อไฟล์ (ไม่แสดงชื่อสกุล) มาให้เลือก โดยเป็นเมนูแบบเลื่อนแถบสว่างบรรทัดละ 7 ไฟล์ ส่วนจำนวนแถวขึ้นอยู่กับจำนวนไฟล์ที่มีอยู่

❖ ให้เลื่อนแถบสว่างด้วยแป้นลูกศร แป้น [Home] และ [End] เพื่อไปยังไฟล์ข้อมูลที่ต้องการ

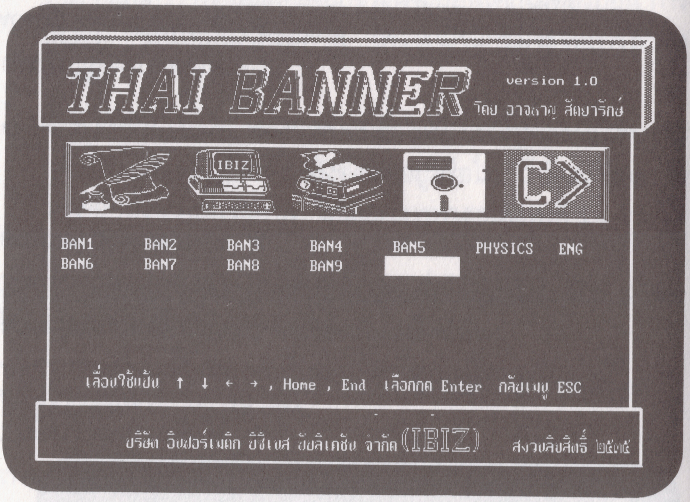
❖ เมื่อต้องการอ่านแฟ้มข้อมูลใดก็ให้กดแป้น [Enter] เมื่อแถบสว่างอยู่ ณ ชื่อไฟล์นั้น

กรณีที่ท่านเรียกไฟล์ tbanner.exe ผ่านไดรฟ์ A หรือไดรฟ์ B นั้น ก่อนจะกดแป้น [Enter] จะต้องแน่ใจว่ามีไฟล์แฟ้มภาพ tbanner1.pic และ tbanner2.pic อยู่ในไดรฟ์ A หรือ B ด้วย มิฉะนั้นจะทำให้ป้ายประกาศที่อ่านมานั้นแสดงผลบนจอภาพไม่ตรงกับที่เขียนไว้

❖ หากไม่มีชื่อไฟล์ในสารบัญที่กำลังติดต่ออยู่ ก็จะต้องเปลี่ยนสารบัญ-ไฟล์ให้กดแป้น [Esc] เพื่อกลับไปเลือกรายการเปลี่ยนสารบัญไฟล์ใหม่

เมื่อเปลี่ยนสารบัญไปเป็นสารบัญที่ต้องการแล้วก็ให้ย้อนกลับไปทำตามลำดับที่ 1 ใหม่

ในช่วงที่มีการเลือกไฟล์ของแฟ้มข้อมูลป้ายประกาศนั้น จะมีลักษณะเป็นดังรูปที่ 5.5



รูปที่ 5.5 *เมนูแสดงชื่อไฟล์ในการเลือกไฟล์เพื่ออ่านข้อมูลมาใช้งาน*

ข้อควรคำนึงในการอ่านและเขียนแฟ้มข้อมูล

ในกรณีที่ท่านไม่มีฮาร์ดดิสก์ใช้ จะมีข้อพึงระวังในการอ่านหรือเขียนแฟ้มข้อมูลอยู่ประการหนึ่งคือ จะต้องใส่แผ่นโปรแกรม Thai Banner ไว้ในไดรฟ์ที่รันไฟล์ tbanner.exe ตลอดเวลา ทั้งนี้เพราะ

❖ ถ้าป้ายประกาศที่อ่านจากแฟ้มข้อมูลนั้นมีการลงภาพ โปรแกรมจะต้องอ่านแฟ้มภาพตามข้อมูลเกี่ยวกับภาพที่ได้กำหนดไว้มาสู่หน่วยความจำ ทั้งนี้เพราะโปรแกรมได้ออกแบบให้การอ่านแฟ้มภาพของแต่ละป้ายข้อมูลนั้นอ่านมาเพียงครั้งเดียวเท่านั้น จึงต้องอ่านข้อมูลของแฟ้มภาพมาทันที

- ❖ หากไม่มีแฟ้มภาพให้อ่าน โปรแกรมจะไม่แจ้งข้อผิดพลาดให้ทราบ แต่เมื่อดูป้ายประกาศนั้น ๆ บนจอภาพ จะไม่มีภาพที่ถูกต้องปรากฏบนจอภาพ หรืออาจไม่มีภาพให้ ดังนั้นหากปรากฏความผิดพลาดแบบนี้ ให้ตระหนักว่าท่านไม่ได้ใส่แผ่นโปรแกรมในไดรฟ์ที่เข้ามายังโปรแกรม Thai Banner นี้ ท่านจะต้องใส่แผ่นในช่องอ่าน แล้วทำการอ่านแฟ้มป้ายประกาศนั้น ๆ เข้ามาใหม่ ตัวอย่างเช่น

- ❖ ใส่แผ่นโปรแกรม Thai Banner ในไดรฟ์ A แล้วรันไฟล์ tbanner.exe A>tbanner

- ❖ ในกรณีนี้ห้ามนำแผ่นโปรแกรม Thai Banner ออกจากไดรฟ์ A โดยเด็ดขาด

- ❖ ในกรณีนี้ท่านสามารถอ่านหรือเขียนแฟ้มข้อมูลป้ายประกาศลงมายังไดรฟ์ A ได้ทุกกรณี

- ❖ เปลี่ยนสารบัญไฟล์ไปใช้ไดรฟ์ B โดยใส่แผ่นข้อมูลไว้ในไดรฟ์ B นั้นด้วย บนจอภาพเมนูหลักจะแสดง B> ที่รายการสุดท้าย ซึ่งหมายความว่าการอ่านเขียนแฟ้มข้อมูลของป้ายประกาศจะกระทำกับแผ่นบันทึกในไดรฟ์ B เท่านั้น

- ❖ เมื่อมีการสร้างป้ายและมีการลงภาพ โปรแกรมจะไปอ่านแฟ้มภาพจากแผ่นบันทึกในไดรฟ์ A เท่านั้น ดังนั้นจึงต้องใส่แผ่นโปรแกรม Thai Banner ไว้ที่ไดรฟ์ A ตลอดเวลาดังกล่าว

- ❖ ในกรณีที่เครื่องคอมพิวเตอร์ของท่านมีเพียงไดรฟ์เดียวก็แก้ปัญหาได้โดยการ copy ไฟล์แฟ้มภาพ tbanner1.pic และ tbanner2.pic มาใส่ในแผ่นบันทึกที่ใช้ในการเก็บแฟ้มข้อมูล แล้วนำไปใส่แทนแผ่นโปรแกรม Thai Banner เมื่อเข้าสู่เมนูหลักของโปรแกรมแล้วก็ได้

❖ สำหรับเครื่องที่มีฮาร์ดดิสก์ก็ควรนำโปรแกรม Thai Banner ทั้งหมดไปใส่ในฮาร์ดดิสก์ แล้วรันผ่านฮาร์ดดิสก์ก็จะไม่มีปัญหาในเรื่องของแฟ้มภาพ เพราะโปรแกรม Thai Banner จะจำเองว่าต้องอ่านแฟ้ม-ภาพจากสารบัญไฟล์ใดโดยอัตโนมัติ

6
ความรู้เพิ่มเติมอื่น ๆ

สำหรับโปรแกรม Thai Banner ที่ท่านกำลังใช้งานอยู่นี้ มีสิ่งที่ควรรู้เพิ่มเติมจากที่ได้อธิบายการใช้งานไปแล้วดังนี้

การสร้างภาพเพิ่มเติม

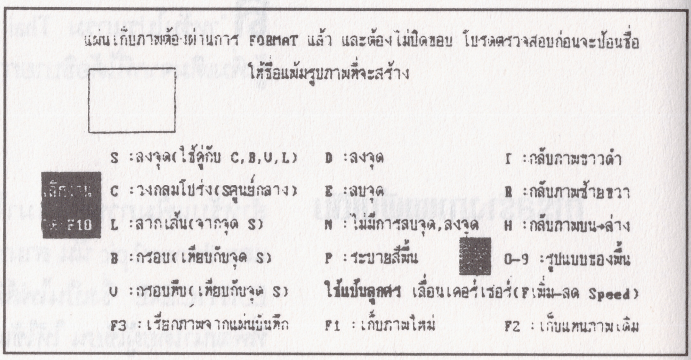
สำหรับแฟ้มภาพที่จัดมาให้จำนวน 2 แฟ้มภาพ คือ tbanner1.pic และ tbanner2.pic นั้น สามารถสร้างภาพเพิ่มเติมได้โดยใช้โปรแกรมชื่อ EDITPIC.EXE ซึ่งเป็นไฟล์ที่อยู่ในชุดโปรแกรมกราฟิก "THAISHOW" ที่พัฒนาโดยผู้เขียน ให้ใช้แผ่นที่สองของชุดซึ่งมีชื่อแผ่นว่า โปรแกรมสนับสนุน "THAISHOW" (ถ้าสนใจโปรดติดต่อที่ผู้เขียน)

โดยโปรแกรมนี้สามารถวาดภาพได้ดังนี้

- สภาวะการลงจุด โดยเมื่อเลื่อนเคอร์เซอร์ไปจะมีการลงจุด ณ จุดที่เคอร์เซอร์เคลื่อนย้ายไป
- สภาวะการลบจุด โดยเมื่อเลื่อนเคอร์เซอร์ไปจะมีการลบจุด ณ จุดที่เคอร์เซอร์เคลื่อนย้ายไป
- วาดรูปวงกลมได้

- ❖ สร้างกรอบสี่เหลี่ยมได้ทั้งกรอบทึบและกรอบโปร่ง
- ❖ ระบายพื้นได้
- ❖ กำหนดภาพกลับขาวดำ กลับซ้ายขวา กลับบนล่างได้
- ❖ เรียกภาพจากแฟ้มอื่นเข้ามาแก้ไขหรือผสมกับภาพที่กำลังเขียนได้ ฯลฯ

เมื่อรันไฟล์ editpic.exe จะมีรายการทำงานดังรูปที่ 6.1



รูปที่ 6.1 *ภาพแสดงรายการเลือกในไฟล์ editpic.exe*

การติดตั้งลงฮาร์ดดิสก์

ถ้าต้องการใช้งานกับฮาร์ดดิสก์ท่านสามารถทำได้โดยง่าย เพราะโปรแกรมไม่ได้มีการตรวจสภาวะข้างเคียงของฮาร์ดดิสก์แต่อย่างใด จึงสามารถใช้งานร่วมกับฮาร์ดดิสก์ได้

สมมติว่าเราใช้ไดรฟ์ C และสร้างไดเรกทอรีชื่อ TBANNER จะมีขั้นตอนการติดตั้งดังนี้

❖ เปลี่ยนไดเรกทอรีมาที่ C:\

❖ สร้างไดเรกทอรีสำหรับการเก็บโปรแกรม Thai Banner ด้วยคำสั่ง MD คือ

  C:\>md tbanner ตามด้วยการกดแป้น [Enter]

❖ ใส่แผ่นโปรแกรม Thai Banner ในไดรฟ์ A

❖ ทำการสำเนาไฟล์ในไดรฟ์ A ไปใส่ในไดเรกทอรี C:\TBANNER ดังนี้

  C:\>copy a:*.* c:\tbanner ตามด้วยการกดแป้น [Enter]

จะมีการสำเนาไฟล์ตามคำสั่งที่กำหนด รอสักครู่ก็จะกลับคืนสู่ดอสตามปกติ

❖ สร้างแบตช์ไฟล์เพื่อที่จะทำให้เกิดความสะดวกต่อการใช้งาน สมมติชื่อไฟล์นี้คือ banner.bat จะทำได้ดังนี้

C:\>copy con banner.bat	ตามด้วยการกดแป้น [Enter]
echo off	ตามด้วยการกดแป้น [Enter]
cd\tbanner	ตามด้วยการกดแป้น [Enter]
tbanner	ตามด้วยการกดแป้น [F6]

จะได้ไฟล์ banner.bat อยู่ในไดเรกทอรี C:\ นำไปเรียกใช้งานได้

❖ การเรียกใช้งานก็ให้เรียกผ่านไฟล์ banner.bat ได้ทันที ไม่ว่าในขณะนั้นจะอยู่ในไดเรกทอรีใด

ไฟล์ command.com, autoexec.bat นั้นอาจลบทิ้งได้ เพราะไม่ได้ใช้งานอีกต่อไปแล้วเมื่อมีการรันผ่านฮาร์ดดิสก์

ข้อควรระวัง ก่อนจบส่วนของผู้ใช้และจะไปอธิบายในส่วนที่โปรแกรมเมอร์ควรรู้นั้น ผู้เขียนขอสรุปในบางเรื่องไว้อีกครั้งหนึ่งดังนี้

- สำหรับการใช้งานกับไดรฟ์ A หรือ B นั้น มีข้อที่ต้องระวังอย่างหนึ่ง คือต้องใส่แผ่นที่มีไฟล์ tbanner1.pic และ tbanner2.pic อยู่ในไดรฟ์ที่เรียกใช้ไฟล์ tbanner.exe ด้วยเสมอ ทั้งนี้เพราะว่าในการอ่านแฟ้มภาพนั้นระบบจะอ่านจากไดรฟ์ที่เข้าสู่โปรแกรม Thai Banner เท่านั้น

- ในกรณีที่รันจากฮาร์ดดิสก์ ปัญหานี้จะไม่เกิดขึ้น

- ทางแก้ไขในกรณีที่ท่านต้องการรันผ่านไดรฟ์ A หรือ B ก็คือให้สำเนาแฟ้มภาพทั้งสองแฟ้มนั้นมาเก็บไว้ในแผ่นที่เป็นแผ่นข้อมูลด้วย

- ปัญหาอีกข้อหนึ่งก็คือ ห้ามใช้โปรแกรมพิมพ์เอกสารหรือโปรแกรมเอดิเตอร์ของภาษาต่าง ๆ เรียกแฟ้มข้อมูลที่มีชื่อสกุล .BAN เข้ามาแล้วเก็บทับลงในชื่อเดิม ทั้งนี้เนื่องจากจะไม่สามารถอ่านข้อมูลทั้งหมดเข้ามายังหน่วยความจำของโปรแกรมเหล่านี้ได้ เมื่อเขียนทับในไฟล์เดิมจะมีผลทำให้ป้ายประกาศทั้งหมดหายไป ซึ่งจะมีผลทำให้ไม่สามารถนำไปแสดงในโปรแกรม Thai Banner ได้อีกต่อไป ทั้งอาจมีผลข้างเคียง ทำให้โปรแกรมเกิดความผิดพลาดในการทำงานด้วย

- เนื่องจากไฟล์ tbanner.exe ในแผ่นที่จำหน่ายพร้อมกับคู่มือนั้นได้ใช้ไฟล์ pklite.exe ทำการอัดให้ไฟล์มีขนาดสั้นลง ดังนั้นหากท่านต้องการจะขยายไฟล์นี้ให้คืนกลับมาเป็นขนาดเดิมก็ทำได้โดยใช้ไฟล์ pklite.exe ดังนี้
  C:\>pklite tbanner -x

โดยจะต้องมีเนื้อที่ในแผ่นบันทึก (กรณีที่ทำผ่านไดรฟ์ A หรือ B) เหลือพอที่จะทำการขยายไฟล์นี้ด้วย

- คำอธิบายสำหรับผู้ใช้โปรแกรม Thai Banner คงจะมีเพียงแค่นี้ ในบทต่อไปนี้จะเป็นแนวคิด หลักการ และโปรแกรมในการทำงานเพื่อให้โปรแกรมเมอร์ได้รับรู้ในความคิดที่ผู้เขียนใช้พัฒนาโปรแกรม Thai Banner นี้

- ถ้าท่านไม่มีความรู้เรื่องการเขียนโปรแกรมก็คงต้องจบการอ่านเพียงแค่นี้แหละนะ!

การสร้างฟอนต์แบบเวกเตอร์ 7

ในช่วงแรกของหนังสือเล่มนี้ได้กำหนดไว้สำหรับผู้ใช้โปรแกรมเท่านั้น แต่เพื่อให้โปรแกรมนี้มีความสมบูรณ์ยิ่งขึ้น ผู้เขียนจึงได้นำส่วนของโปรแกรมที่เป็นรากฐานของโปรแกรม Thai Banner นี้มาเขียนถึงเพื่อที่จะเป็นแนวทางให้โปรแกรมเมอร์ได้มองเห็นจุดที่จะพัฒนาสร้างเสริมให้ได้โปรแกรมที่มีรูปแบบแปลกตา ตลอดจนนำไปใช้งานร่วมกับซอฟต์แวร์ตัวอื่น ๆ ได้

ดังนั้นตั้งแต่บทนี้เป็นต้นไปคงเป็นส่วนสำหรับโปรแกรมเมอร์โดยเฉพาะ โดยที่ผู้เขียนได้พัฒนาโปรแกรม Thai Banner บนเทอร์โบปาสคาล จึงขอนำเสนอในรูปของภาษาปาสคาลทั้งหมด

โครงสร้างของไฟล์ banner.fon

ปัญหาแรกที่ผู้เขียนมองเห็นในโปรแกรม Thai Banner นี้ก็คือเรื่องของตัวอักขระทั้งภาษาอังกฤษและภาษาไทย เพราะว่าต้องมีการทำงานที่สำคัญคือ

❖ ต้องนำมาขยายให้โตขึ้นได้โดยยังคงรูปร่างเดิมไม่ผิดเพี้ยน ซึ่งแน่นอนการเก็บอักขระแบบทั่ว ๆ ไปคงจะทำไม่ได้ กล่าวคือ ยิ่งขยายก็จะได้อักขระที่มีรายละเอียดน้อยลง

❖ ต้องนำอักขระชุดเดียวกันนี้มาแสดงได้หลายรูปแบบคือ สามารถสร้างตัวเอียง แสดงแนวเฉียงขึ้นหรือเฉียงลงได้ ฯลฯ เป็นต้น

❖ ต้องพิมพ์ลงกระดาษด้วยเครื่องพิมพ์ชนิดหัวเข็มตามรูปแบบที่ถูกต้องได้ ไม่ว่าข้อความนั้น ๆ จะมีความยาวเท่าใดก็ตาม (ข้อความที่ยาวมาก ๆ จะไม่สามารถแสดงให้จบในหนึ่งจอภาพได้)

ยุทธวิธีการสร้างอักขระ

ปัญหาแรกที่ผู้เขียนเผชิญก็คือ การสร้างอักขระแบบใดจึงจะตรงกับความต้องการ ซึ่งเป็นปัญหาที่ต้องใช้ความคิดมากทีเดียว

ในที่สุดผู้เขียนก็ตกลงใจว่าจะสร้างอักขระเหล่านี้อย่างไร โดยแนวคิดที่ได้มาก็คือ

❖ กำหนดการสร้างอักขระแบบกำหนดตำแหน่งต่าง ๆ เป็นคู่ลำดับ (x,y) แบบกราฟ โดยถือว่าจุดที่ตามหลังจะมีตำแหน่งสัมพันธ์กับจุดที่ลงก่อนหน้าเสมอ ซึ่งก็คือการสร้างอักขระแบบเวกเตอร์แบบหนึ่งนั่นเอง สำหรับขนาดของตัวอักขระได้กำหนดเป็น 40X36 จุด ส่วนอักขระที่เป็นสระก็ได้มีการลดส่วนสูงลง

❖ กำหนดคู่ลำดับเริ่มต้นให้กับอักขระนั้น ๆ เก็บไว้เป็นข้อมูลชุดแรก คู่ลำดับนี้มีไว้เพื่อให้แสดงตำแหน่งของสระต่าง ๆ ให้ถูกต้องตามความเป็นจริง และช่วยให้แสดงอักขระแบบจัดระยะห่างสวยงามขึ้น

❖ ข้อมูลที่เป็นคู่ลำดับของอักขระแต่ละตัวจะสิ้นสุดเมื่อคู่ลำดับนั้นมีค่า x=86

❖ ถ้าคู่ลำดับใดของอักขระนั้นมีค่า x=86 หรือ x=99 จะวาดรูปอักขระนั้นครั้งหนึ่ง

❖ ถ้าคู่ลำดับใดของอักขระนั้นมีค่าx=99 แสดงว่ายังมีการวาดรูปอักขระนั้นต่อ แต่จะมีการเริ่มต้น ณ จุดใหม่ จึงให้เลื่อนไปตรวจคู่ลำดับถัดมาเพื่อสร้างอักขระส่วนอื่นต่อ ดังนี้

1. ถ้าคู่ลำดับถัดมามีค่า x<>127 ก็จะเป็นการวาดรูปด้วยพื้นแบบเดิมที่กำหนดให้ตัวอักขระนั้น

2. ถ้าคู่ลำดับถัดมามีค่า x=127 ก็จะเป็นการวาดรูปด้วยพื้นสีดำ แนวคิดนี้จะมีประโยชน์ในการวาดรูปส่วนหัวของอักขระภาษาไทย หรือการวาดเชิงของอักขระบางตัว เช่น ญ ฐ เป็นต้น

❖ จำนวนของคู่ลำดับของแต่ละอักขระไม่ได้กำหนดไว้ให้เท่ากัน จะมากหรือน้อยขึ้นอยู่กับลักษณะของตัวอักขระเอง แต่จะไม่เกิน 254 คู่ลำดับ

❖ มีตัวแปรเพื่อให้หาจุดเริ่มต้นของแต่ละอักขระได้อย่างรวดเร็ว

ดังนั้นการสร้างอักขระในรูปแบบที่เรียกว่า ตารางรูป (shape table) จึงเหมาะสำหรับการนำมาใช้ ส่วนการแสดงตัวอักษรนั้นผู้เขียนเรียกใช้วิธีกำหนดรูปเหลี่ยมเพื่อใช้กับคำสั่ง FillPoly

ส่วนต่าง ๆ ของไฟล์ข้อมูล (ผู้เขียนให้ชื่อไฟล์ว่า banner.fon) ดังตารางที่ 7.1

ตารางที่ 7.1 *แสดงโครงสร้างการเก็บแฟ้มข้อมูลอักขระ*

ShapeNo 1 byte	PointStart 2 bytes	SizeOfShape 2 bytes	ShapeHead 510 bytes	ShapePoly ยาวเท่ากับ SizeOfShape

ผังแสดงลำดับของตัวแปรที่เก็บในไฟล์ banner.fon

ผังของตัวแปร ShapeHead เป็นดังนี้

เก็บตำแหน่งเริ่มต้น รูปที่ 1 ขนาด 2 ไบต์	เก็บตำแหน่งเริ่มต้น รูปที่ 2 ขนาด 2 ไบต์	...	เก็บตำแหน่งเริ่มต้น รูปที่ 255 ขนาด 2 ไบต์

โดยตัวแปรที่เก็บค่าต่าง ๆ จะมีดังนี้

- ❖ ShapeNo เก็บลำดับที่ของตัวอักขระที่จะสร้างเพิ่ม
- ❖ PointStart เก็บจุดเริ่มต้นของอักขระที่จะสร้างเพิ่ม
- ❖ SizeOfShape เก็บขนาดความยาวของข้อมูลที่ใช้สร้างอักขระทั้งหมด
- ❖ ShapeHead เก็บตำแหน่งเริ่มต้นของอักขระแต่ละตัว
- ❖ ShapePoly ข้อมูลของอักขระทั้งหมดที่ได้จากการสร้าง

กำหนดตัวแปรใช้งาน

เมื่อได้กำหนดรูปแบบไว้แล้วก็มาถึงการเขียนโปรแกรมเพื่อสร้างอักขระตามแนวคิดนี้ โดยได้กำหนดตัวแปรไว้ดังนี้

- ❖ สร้างรูปแบบของพื้นอีก 10 แบบด้วยตัวแปร PatternMe
- ❖ กำหนดชนิดของตัวแปรใหม่ในชื่อ PolygonType เพื่อใช้รองรับรูปเหลี่ยมของอักขระที่เราสร้าง

❖ กำหนดตัวแปร PolyByte; PolyInt; เพื่อนำมารองรับคู่ลำดับที่เก็บในไฟล์ banner.fon และคู่ลำดับที่ได้จากการคำนวณเมื่อนำไปสร้างรูปอักขระบนจอภาพ

โดยมีรูปแบบคำสั่งดังรูปที่ 7.1 ซึ่งมีชุดคำสั่ง LoadFile; SetPattern; ClsGr; เพื่อใช้งานต่าง ๆ คือ เรียกไฟล์ กำหนดรูปแบบพื้น และลบจอภาพ

```
PROGRAM EDIT_SHAPETABLE_THAI_BANNER;
uses crt,dos,graph;
type PolygonType=array[1..255] of pointtype;
     PolyByte=array[1..2] of shortInt;
     PolyInt=array[1..2] of Integer;
const patternMe:array[0..9] of FillpatternType=
   (($FE,$EF,$FE,$EF,$FE,$EF,$FE,$EF),
   ($18,$3C,$7E,$FF,$FF,$7E,$3C,$18),
   ($77,$77,$77,$77,$77,$77,$77,$77),
   ($95,$59,$95,$59,$95,$59,$95,$59),
   ($E7,$7E,$E7,$7E,$E7,$7E,$E7,$7E),
   ($6D,$D6,$6D,$D6,$6D,$D6,$6D,$D6),
   ($B3,$3B,$B3,$3B,$B3,$3B,$B3,$3B),
   ($5C,$FF,$C3,$C3,$C3,$C3,$FF,$5C),
   ($5C,$E7,$7E,$C5,$E7,$7E,$C5,$E7),
   ($18,$3C,$7E,$FF,$7E,$5A,$18,$18));
var Monoscreen:boolean;  SizeOfShape,PointStart:word;
   PicCursor:array[0..2] of pointer;
   Xpix,Ypix,XStart,YStart,X_Shape,Y_shape:word;
   NoPoly,ShapeNo,ColorBg,ColorBox,PattNo,Lineck:byte;
   PolyPoint:array[1..255] of PolyInt;
```

```
    regs:Registers;   ch,Funckey:char;
    ShapeHead:array[1..255]  of  Word;
    ShapePoly:array[1..20000]  of  PolyByte;

PROCEDURE LoadFile;
var f:file;
begin
    assign(f,'Banner.fon');  reset(f,1);
    blockread(f,ShapeNo,1);
    blockread(f,PointStart,2);
    blockread(f,SizeOfShape,2);
    blockread(f,ShapeHead,510);
    blockread(f,ShapePoly,SizeOfShape);
   Close(f);
end;
PROCEDURE SetPattern;
begin   setcolor(ColorBox);
    if Monoscreen then
      begin
          ColorBg:=1;  setcolor(1);
      end;
    case PattNo of
        0:setfillstyle(1,0);
        1..10:setfillpattern(patternMe[PattNo-1],ColorBg);
        11..20:setfillstyle(PattNo-10,ColorBg);
    end;
end;
PROCEDURE ClsGr  (x1,y1,x2,y2:word);
begin  setfillstyle(1,0);
```

```
        Bar(x1,y1,x2,y2);
        setfillstyle(1,1);
end;

PROCEDURE OpenGraph;
var
    GrDriver,GrMode :integer;
begin
    MonoScreen:=False;
    detectgraph(Grdriver,Grmode);
    case Grdriver of
         7:begin MonoScreen:=true;
                 GrMode:=0;
           end;
    2,3,5,9:GrMode:=1;
    end;
    InitGraph(GrDriver, GrMode, '');
end;
```

รูปที่ 7.1 *โปรแกรมเริ่มต้นแสดงตัวแปรที่ใช้และชุดคำสั่ง 2 ชุดคำสั่ง*

ส่วนรูปที่ 7.2 นั้นเป็นชุดคำสั่งในการเก็บไฟล์ชื่อ SaveFile และ KeyPress ซึ่งจะใช้เฉพาะตอนที่สร้างอักขระเท่านั้น

```
PROCEDURE SaveFile;
var f:file;
begin
    assign(f,'Banner.fon');
    Rewrite(f); reset(f,1);
    blockwrite(f,ShapeNo,1);
```

```
        blockwrite(f,PointStart,2);
        blockwrite(f,SizeOfShape,2);
        blockwrite(f,ShapeHead,510);
        blockwrite(f,ShapePoly,SizeOfShape);
        Close(f);
end;
PROCEDURE  KeyPress(X,Y:word);
var  k:word;
begin
    if  x<>0  then
      begin
          getimage(x-3,y-1,x+4,y+8,PicCurSor[0]^);  k:=0;
          repeat
    if  k=0  then   PutImage(x-3,y-1,PicCursor[0]^,4);
    Inc(k);
    if  k=200  then  PutImage(x-3,y-1,PicCursor[0]^,0);
    if  k=400  then  k:=0;  delay(3);
  until  keypressed;  PutImage(x-3,y-1,PicCursor[0]^,0);
end;
  Funckey:=#0;       Regs.AH:=$0;  intr($16,regs);
  ch:=chr(regs.AL);  If  ch=#0  then  Funckey:=Chr(regs.AH);
end;
```

รูปที่ 7.2 *ชุดคำสั่งสำหรับเก็บไฟล์และแสดงเคอร์เซอร์กราฟิก*

การสร้างรูปอักขระ

เพื่อความสะดวกในการสร้างรูปอักขระเก็บลงไฟล์ banner.fon จึงกำหนดให้มีการลากเส้นบนจอภาพ ซึ่งทำให้มองเห็นรูปของอักขระที่กำลังสร้างได้ว่ามีรูปตามความต้องการของเราหรือไม่ ดังนั้นจึงจำเป็นต้องเขียนชุดคำสั่งขึ้นมา

โปรแกรมจึงประกอบด้วย

❖ การสร้างตารางแสดงขอบเขตของตัวอักษรบนจอภาพ

❖ แสดงข้อความให้ทราบว่ามีการใช้แป้นพิมพ์ใดบ้างในการสร้างรูปอักขระ

❖ แสดงรูปอักขระที่ได้สร้างไว้แล้วนั้นให้ทราบว่าถ้ามีขนาดโตขึ้นจะมีความพอใจหรือไม่ เพื่อให้ตัดสินใจว่าจะเก็บภาพนี้หรือจะสร้างใหม่

❖ มีคำสั่งการทำงานต่าง ๆ เมื่อได้กดแป้นพิมพ์ตามที่กำหนดไว้ในอันดับที่ 2

สำหรับแป้นพิมพ์ที่ใช้ในการสร้างรูปอักขระนั้นประกอบด้วยแป้นพิมพ์ต่าง ๆ ดังนี้

❖ แป้นลูกศร 8 แป้นใช้สำหรับเลื่อนตำแหน่งของเคอร์เซอร์ไปยังจุดที่เราต้องการ

❖ แป้นแคร่ยาว (space bar) ใช้สำหรับเลือกจุด ณ ตำแหน่งที่เคอร์เซอร์อยู่เป็นคู่ลำดับของตัวอักขระ

❖ แป้น [Ins] ใช้เมื่อต้องการจะให้การแสดงรูปอักขระจบลงไปช่วงหนึ่งก่อน แล้วไปเริ่มต้นเขียนตัวอักขระนี้ใหม่ในตำแหน่งที่กดแป้นแคร่ยาวถัดไป ได้กำหนดให้มีเสียงดังเมื่อกดแป้นนี้ (ได้คู่ลำดับมีค่า 99, 99)

❖ แป้น [Del] ใช้ตามหลังการกดแป้น [Ins] และได้เลื่อนเคอร์เซอร์ไปยังจุดเริ่มต้นใหม่ (ก่อนกดแป้นแคร่ยาวครั้งถัดไป และต้องกดตามหลังการกดแป้น [Ins] เท่านั้น) จะช่วยให้การวาดรูปอักขระในช่วงถัดไปเป็นการวาดด้วยพื้นสีดำ (ได้คู่ลำดับมีค่าเป็น 127,127) มีเสียงดังเมื่อกดแป้นนี้ด้วย

ลำดับการทำงานคือ กดแป้น [Ins]-[==>] เลื่อนเคอร์เซอร์ [==>] กดแป้น [Del] ซึ่งจะทำให้สามารถวาดรูปส่วนหัวของอักขระ เช่น อ ป พ ห ฯลฯ ให้มีความสวยงามขึ้น

❖ แป้น [Enter] ใช้เมื่อสิ้นสุดการสร้างอักขระนั้น ๆ (คู่ลำดับเป็น 86, 86)

สำหรับโปรแกรมการสร้างรูปอักขระในส่วนของการเลื่อนเคอร์เซอร์และลงคู่ลำดับจะเป็นดังรูปที่ 7.3

```
PROCEDURE SelectXy(x,y:integer);
var   k,Xold,Yold,Xshape,YShape:word;
      c,CC:byte; xx,yy:Integer; STT:string[2];
  begin
    Xold:=x; Yold:=y; str(NoPoly-1,STT);
    XShape:=X_shape; YShape:=Y_shape;
    ClsGR(198,142,300,170);
    outtextxy(200,150,'Point No.'+STT);
    getimage(x-3,y-3,x+3,y+3,PicCurSor[0]^);
    ClSGr(x-3,y-3,x+3,y+3);
    line(x-3,y,x+3,y);    line(x,y-3,x,y+3);
    c:=10; cc:=0;
    if Monoscreen then begin c:=1; cc:=4; end;
    getimage(x-3,y-3,x+3,y+3,PicCurSor[1]^);
    getimage(X_Shape-1,Y_Shape-1,X_shape+1,
                                   Y_Shape+1,PicCurSor[2]^);
  repeat
    repeat k:=0;
      repeat
```

```
if k=0 then
      begin PutImage(x-3,y-3,PicCursor[1]^,CC);
            putpixel(X_Shape,Y_Shape,c);
      end;
      Inc(k);
   if k=200 then
      begin PutImage(x-3,y-3,PicCursor[0]^,0);
            putpixel(X_Shape,Y_Shape,0);
      end;
   if k=400 then k:=0; delay(3);
 until keypressed; KeyPress(0,0);
until (Funckey in [#71..#73,#75,#77,#79..#83]) or
                              (Ch in [#13,#32]);
PutImage(x-3,y-3,PicCursor[0]^,0);
PutImage(X_Shape-1,Y_Shape-1,PicCursor[2]^,0);
case Funckey of
   #71:begin Dec(y,8); Dec(x,8);
             Dec(Y_shape); dec(X_shape);
       end;
   #72:begin Dec(y,8); Dec(Y_shape); end;
   #73:begin Dec(y,8); Inc(x,8);
             Dec(Y_shape); Inc(X_Shape);
       end;
   #75:begin Dec(x,8); Dec(X_shape); end;
   #77:begin Inc(x,8); inc(X_shape); end;
   #79:begin Inc(y,8); Dec(x,8);
             inc(Y_shape); dec(X_shape);
       end;
   #80:begin Inc(y,8); Inc(Y_shape); end;
   #81:begin Inc(y,8); Inc(x,8);
```

```
                inc(Y_shape); inc(X_shape);
            end;
end;
    if y<8      then begin y:=8;   inc(Y_shape); end;
    if y>312    then begin y:=312; dec(Y_Shape); end;
    if x<10     . then begin x:=10;  inc(X_shape); end;
    if x>170    then begin x:=170; dec(X_shape); end;
    getimage(x-3,y-3,x+3,y+3,PicCurSor[0]^);
    getimage(X_Shape-1,Y_Shape-1,X_shape+1,
                               Y_Shape+1,PicCurSor[2]^);
  if FuncKey=#82 then
    begin   Sound(450);  delay(500);
            Lineck:=27;  Xold:=XStart; YOld:=YStart;
            ch:=#0;
            PolyPoint[NoPoly][1]:=99;
            PolyPoint[NoPoly][2]:=99;  NoSound;
            Inc(NoPoly);
    End;
  if (FuncKey=#83) and (PolyPoint[NoPoly-1][1]=99) hen
    begin  Sound(500);  delay(500);
           PolyPoint[NoPoly][1]:=127;
           PolyPoint[NoPoly][2]:=127;
           Nosound; Inc(NoPoly);
    End;
  until ch in [#13,#32];
      Funckey:=#0;  PutPixel(x,y,c);
      xx:=x-Xold;  yy:=Yold-y;
    if ch=#32 then
      begin
       if  (NoPoly>2) and (LineCk=0) then      begin
```

```
            setlineStyle(0,$FFFF,3);   line(Xpix,Ypix,x,y);
            line(XShape,Yshape,X_shape,Y_shape);  end;
            LineCk:=0;
            PolyPoint[NoPoly][1]:=xx  div 8;
            PolyPoint[NoPoly][2]:=yy  div  8;
      end;
        Xpix:=x; YPix:=y;
        XShape:=X_shape; YShape:=Y_shape;
   end;
```

รูปที่ 7.3 *ชุดคำสั่งในการสร้างอักขระเพื่อนำไปเก็บในไฟล์ tbanner.fon*

คำอธิบายสำหรับชุดคำสั่ง SelectXY นี้เป็นการเลื่อนจุดในการวาดภาพที่โตกว่าความเป็นจริง 8 เท่า เพื่อที่จะได้สะดวกในการสร้างรูปอักขระนั้น ๆ ดังนั้นขนาดบนจอภาพจึงเป็นขนาด 8 เท่าของอักขระที่เก็บ โดยได้แสดงรูปอักขระตามความเป็นจริงไว้ที่ส่วนขวาของตารางด้วย รายละเอียดเกี่ยวกับการทำงานมีดังนี้

❖ กำหนดจุดเริ่มต้นของจุดก่อนที่จะมีการเลื่อนเคอร์เซอร์ เพื่อใช้หาระยะทางที่สัมพันธ์กับจุดที่จะเลื่อนไป (ตัวแปร Xold, Yold)

❖ แสดงข้อความให้รู้ว่าเป็นคู่ลำดับที่เท่าใดของอักขระนั้น ๆ

❖ สร้างเคอร์เซอร์รูปกากบาทและเก็บภาพของเคอร์เซอร์ไว้

❖ แสดงเคอร์เซอร์แบบกะพริบ พร้อมกับแสดงจุดสว่าง ณ ตำแหน่งที่แสดงรูปอักขระเท่าของจริง

❖ เมื่อกดแป้นลูกศร (Funckey =#71-#73, #75, #77 #79-#81) ให้เลื่อนเคอร์เซอร์ตามค่า x, y ที่คำนวณได้พร้อมกับเลื่อนจุดสว่างตามไป

ในทิศทางเดียวกันด้วยตัวแปร X_shape, Y_Shape พร้อมกับตรวจสอบขอบเขตของตัวแปรเหล่านี้

❖ เมื่อกดแป้น [Ins](Funckey=#82) ก็จะกำหนดคู่ลำดับเป็นค่า 99, 99 เพื่อใช้แยกการวาดรูปออกเป็นส่วน ๆ เช่น การวาด สระ แ เป็นต้น

❖ เมื่อกดแป้น [Del](Funckey=#83) และเป็นการกดต่อจากการกดแป้น [Ins] โดยเลื่อนเคอร์เซอร์ไปยังจุดเริ่มต้นใหม่ จะได้คู่ลำดับมีค่าเป็น 127,127 ซึ่งทำให้การวาดรูปในส่วนถัดไปเป็นพื้นสีดำ

❖ ถ้ากดแป้น [Enter] หรือเป้นแคร่ยาว ก็จะเป็นการออกจากชุดคำสั่งนี้ โดยถ้ากดแป้นแคร่ยาวก็จะเป็นการหาคู่ลำดับแบบสัมพันธ์กับจุดเดิม เพื่อเก็บลงในตัวแปร PolyPoint พร้อมกับลากเส้นให้เห็นรูปตัวอักขระบนจอภาพทั้งในตารางและในส่วนที่เป็นขนาดเท่าของจริงที่จะเก็บลงไฟล์

❖ ถ้ากดแป้น [Enter] จะได้คู่ลำดับมีค่าเป็น 86,86

❖ การที่กำหนดค่าเงื่อนไขให้เป็น 86, 99, 127 ได้เพราะความกว้างของอักขระในแกน x มีค่าสูงสุดเพียง 40 เท่านั้น และค่าของคู่ลำดับมีได้ไม่เกิน 127

ชุดคำสั่งนี้จะถูกเรียกใช้งานจากชุดคำสั่งในรูปที่ 7.6 ที่อยู่ถัดไป

การเก็บอักขระลงไฟล์และการแสดงผลบนจอภาพ

เมื่อเราได้คู่ลำดับต่าง ๆ ของอักขระที่วาดก็นำมาเก็บลงในตัวแปร ShapePoly โดยใช้ชุดคำสั่งชื่อ SavePoly เริ่มด้วยลำดับที่ตามค่าของตัวแปร PointStart ของอักขระนั้น ๆ

ส่วนการนำข้อมูลของอักขระมาแสดงก็สามารถทำได้ดังที่ปรากฏในรูปที่ 7.4 คือ ชุดคำสั่ง ShowPoly

```
PROCEDURE SavePoly(No:byte);
var b:integer;
begin
   if No=1 then
      begin
         SizeOfShape:=1; PointStart:=1;
      end;
      ShapeHead[No]:=PointStart;   b:=0;
   repeat
      Inc(B);
      ShapePoly[PointStart][1]:=PolyPoint[B][1];
      ShapePoly[PointStart][2]:=PolyPoint[B][2];
      Inc(SizeOfShape,2); Inc(PointStart);
   Until PolyPoint[B-1][1]=86;
end;
PROCEDURE ShowPoly(Xpix,Ypix:integer; size:byte);
var
      Poly : PolygonType;
      x,y,Xold,YOld:Integer; P,NO,Patt:byte;
   begin
      SetPattern;
      Xpix:=Xpix-Round(POlyPoint[1][1]*size);
      Ypix:=Ypix-Round(POlyPoint[1][2]*size);
      Xold:=xpix; YOld:=ypix;   No:=2;
   Repeat P:=1;
      Repeat
         x:=Xpix+Round(POlyPoint[No][1]*size);
         Poly[p].x :=x; Xpix:=x;
         y:=Ypix-Round(POlyPoint[No][2]*size);
```

```
Poly[p].y :=y; Ypix:=y;
        Inc(NO); Inc(p);
    until PolyPoint[No][1] in [86,99];
      FillPoly(P-1, Poly);
    if PolyPoint[No][1]=99 then
      begin
        Xpix:=Xold; YPix:=Yold;
        Inc(No);        SetPattern;
      if PolyPoint[No][1]=127 then
        begin
          Patt:=PattNo; PattNo:=0; SetPattern;
          PattNo:=Patt; Inc(no);
        end;
      end;
    Until PolyPoint[No][1]=86;
end;
```

รูปที่ 7.4 *ชุดคำสั่งสำหรับเก็บไฟล์และการแสดงอักขระบนจอภาพ*

โดยมีรายละเอียดของชุดคำสั่งต่าง ๆ ดังนี้

ชุดคำสั่ง SavePoly

❖ กำหนดจุดเริ่มต้นของอักขระในตัวแปร ShapePoly ตามตัวแปร PointStart แล้วเก็บไว้ที่ตัวแปร ShapeHead ตามลำดับที่ของอักขระนั้น ๆ

❖ กำหนดค่าของคู่ลำดับลงในตัวแปร ShapePoly ตามลำดับจากจุดที่ 1, 2, 3 ไปจนถึงจุดที่เก็บค่าคู่ลำดับเป็น 86 ไว้ โดยต้องเลื่อนค่าตัวแปร SizeOfShape และ PointStart ไปเป็นค่าใหม่ด้วย

❖ ค่าของคู่ลำดับ ShapePoly นำมาจากคู่ลำดับของ PolyPoint

ชุดคำสั่ง ShowPoly

❖ กำหนดรูปแบบพื้น สีพื้น และสีขอบของตัวอักขระ ตามค่าที่กำหนด โดยการเรียกใช้ชุดคำสั่ง SetPattern ส่วนนี้กำหนดขึ้นมาเพื่อว่าผู้อ่านอาจนำไปแสดงผลในงานกราฟิกอื่น ๆ ได้ด้วย

❖ ใช้ตัวแปร Poly เก็บจุดต่างๆของอักขระที่ต้องการวาด โดยหาจุดเริ่มต้นที่จะแสดงอักขระตามค่าคู่ลำดับที่ 1 (ของตัวแปร PolyPoint) แล้วจำค่านี้ไว้ในตัวแปร Xold และ Yold

❖ หาคู่ลำดับที่ no ของตัวแปร Poly[no].x และ Poly[no].y จากตัวแปร PolyPoint ตั้งแต่ลำดับที่สองไปจนมีค่าคู่ลำดับเป็น 86, 86 หรือเป็น 99, 99

❖ วาดรูปตัวอักขระที่คำนวณได้ด้วยคำสั่ง FillPoly หากต้องการจะได้รูปโปร่ง อาจใช้คำสั่ง DrawPoly แทนก็ได้

❖ ถ้าค่าคู่ลำดับเป็น 99, 99 ก็จะมีการวาดส่วนที่เหลือเพิ่มเติม โดยถ้าคู่ลำดับถัดมาเป็น 127, 127 ก็จะวาดเป็นรูปพื้นสีดำ

❖ จบการวาดรูปอักขระเมื่อค่าคู่ลำดับเป็น 86,86

โปรแกรมหลักการสร้างรูปอักขระ

ชุดคำสั่งที่ผ่านมาต้องอาศัยโปรแกรมหลักในสร้างตาราง กำหนดการเรียกไฟล์และเก็บไฟล์ เลือกจุดเริ่มต้นของอักขระ ฯลฯ ด้วยดังนั้นเราต้องเขียนโปรแกรมหลักอีกชุดหนึ่งเพื่อให้การทำงานมีความสมบูรณ์ยิ่งขึ้น

โปรแกรมจะเป็นดังชุดคำสั่ง EditShape ในรูปที่ 7.5 ซึ่งค่อนข้างยาว ขอให้ลองทำความเข้าใจด้วยตนเองก่อนที่จะอ่านคำอธิบาย

```
PROCEDURE EditShape;
var  No:word; B:byte; STT:string[3];
     SaveF:Boolean;
begin
     ShapeNo:=1;
     outtextxy(20,30,'Do you want to Load file(Y/N)');
     repeat
          KeyPress(260,30); ch:=Upcase(ch);
     until ch in ['N','Y'];
     if ch='Y' then LoadFile;
  Repeat
     ClsGr(0,0,getMaxX,GetMaxY); no:=10;
     setcolor(7); setlineStyle(0,$FF,1);
     repeat
          line(no,8,no,312); Inc(no,8);
     until no>170;
      no:=8; b:=1;
     repeat
        line(10,no,170,no); Inc(no,8); Inc(b);
     until no>312;
     outtextxy(180,50,'A'); outtextxy(180,145,'B');
     outtextxy(180,265,'C');
     setcolor(12); if monoscreen then setcolor(1);
     Line(0,88,180,88); Line(0,232,180,232);
     if pred(monoscreen) then setcolor(14);
     Outtextxy(300,10,'Edit ShapeTable By Arthan Sattaayarak');
     Outtextxy(310,75,'Select Point for Start');
     Outtextxy(0,235,'1'); Outtextxy(175,235,'2,3');
     Outtextxy(360,90,'( 1 ,2 or 3 ==>');
```

```
        Repeat
          KeyPress(500,90);
        Until ch in ['1'..'3'];
      ClsGR(300,50,630,250);
      PolyPoint[1][1]:=0; PolyPoint[1][2]:=0;
case Ch of
    '1':begin xpix:=10; yPix:=232;
          X_shape:=200;  Y_shape:=210;
      end;
    '2':begin  PolyPoint[1][1]:=5;   PolyPoint[1][2]:=-1;
          xpix:=170;  Ypix:=232;
          X_shape:=240;  Y_shape:=210;
      end;
    '3':begin  PolyPoint[1][1]:=5;   PolyPoint[1][2]:=1;
          xpix:=170;   Ypix:=232;
          X_shape:=240;  Y_shape:=210;
      end;
end;
    Outtextxy(310,30,'arrow  key     ==>   move cursor');
    Outtextxy(310,45,'Space  bar     ==>   drawpoly');
    Outtextxy(310,60,'Ins            ==>   pen up');
    Outtextxy(310,75,'Del            ==>   fillpoly with Pattern 0');
    Outtextxy(310,90,'     (use Del after press Ins)');
    Outtextxy(310,105,'Enter         ==>   End of shape');
    setcolor(10); if Monoscreen then setcolor(1);
    str(ShapeNo,STT);
    Outtextxy(200,130,'Shape  Table  No.'+STT);
    NoPoly:=2;   XStart:=Xpix;
    YStart:=Ypix;  LineCk:=0;
repeat
```

```
SelectXy(xpix,ypix);
if Ch<>#13 then  Inc(NoPoly);
if NoPoly=255 then Ch:=#13;
until ch=#13;
PolyPoint[NoPoly][1]:=86; PolyPoint[NoPoly][2]:=86;
PattNo:=16; ColorBg:=13; ColorBox:=11;
if monoscreen then
  begin
    ColorBg:=1; ColorBox:=1;
  end;
line(340,200,340,340); line(240,260,500,260);
ShowPoly(340,260,2); ShowPoly(500,280,3);
Outtextxy(450,180,'Save This Shape (Y/N)');
repeat
   KeyPress(625,180); ch:=Upcase(ch);
 Until ch in ['N','Y']; SaveF:=True;
if Ch='Y' then
begin SavePoly(ShapeNo); Inc(ShapeNo);
ClsGr(400,200,639,348);
Outtextxy(480,210,'  S :Save file');
Outtextxy(480,230,'  E :Edit other Shape');
Outtextxy(480,250,'  Q :Quit');
Outtextxy(480,270,'press Onekey');
repeat
 KeyPress(590,270); ch:=Upcase(ch);
 if ch='S' Then
   begin
     outtextxy(600,210,'<===');
     SaveFile; SaveF:=False;
```

```
            clsgr(598,190,630,220);
        end;
      Until ch in ['Q','E'];
    end;
    if ShapeNo=255 then ch:='Q';
until Ch='Q';
  if SaveF Then
    begin
      Outtextxy(360,300,'Save '#39'Shape table'#39' to disk(Y/N)');
      repeat
          KeyPress(620,300); ch:=Upcase(ch);
          if ch='Y' then SaveFile;
        until Ch in ['N','Y'];
    end;
end;
{====================    ป้ายประกาศ 1    ====================}
begin
  opengraph;
    if pred(monoscreen) then
    begin
      setcolor(14); setfillstyle(1,3);
    end;
    For xpix:=0 to 2  do getmem(PicCursor[xpix],200);
    EditShape;
  closegraph;
end.
```

รูปที่ 7.5 ชุดคำสั่งโปรแกรมหลักของการสร้างรูปอักขระ

รายละเอียดของการสร้างมีดังนี้

❖ ถามว่าจะเรียกไฟล์ banner.fon ของเดิมหรือไม่ ในการเรียกไฟล์นี้ไม่ได้เขียนให้ตรวจสอบข้อผิดพลาดที่เกิดจากการหาไฟล์ไม่พบไว้ หากต้องการใช้ให้เพิ่มคำสั่งเอง

❖ ทำการวาดตารางเพื่อใช้ในการวาดรูปอักขระลงบนจอภาพ แสดงจุดเริ่มต้น 3 จุด โดยจุดที่ 2, 3 อยู่ที่เดียวกัน จุด 2 เลือกเมื่อสร้างสระ ส่วนจุด 3 ใช้เมื่อสร้างสระบนทั้งหมด ส่วนจุด 1 ไว้สำหรับสร้างอักขระอื่นๆ

❖ ให้เลือกจุดเริ่มต้นในการสร้างอักขระ โดยกดแป้นตัวเลข 1, 2 หรือ 3 ซึ่งจะกำหนดค่าตัวแปร PolyPoint[1][1] และ PolyPoint[1][2] ให้ตรงกับความต้องการ
ตัวแปร PolyPoint[1][1] สำหรับกำหนดตำแหน่งในแกน x
ตัวแปร PolyPoint[1][2] สำหรับกำหนดตำแหน่งในแกน y

❖ แสดงข้อความให้ทราบว่ามีแป้นพิมพ์ใดบ้างที่ใช้งาน พร้อมกับแสดงให้ทราบว่ากำลังสร้างรูปอักขระหมายเลขใด และเป็นคู่ลำดับที่เท่าใด

❖ เลือกใช้ชุดคำสั่ง SelectXY เพื่อสร้างรูปอักขระและจบการสร้างด้วยการกดแป้น [Enter]

❖ กำหนดจุดสิ้นสุดของรูปอักขระ และแสดงรูปของอักขระที่สร้างในขนาดที่ใหญ่กว่าเดิม 2 และ 3 เท่า

❖ ถามว่าพอใจกับรูปอักขระที่สร้างนี้หรือไม่ ถ้ากด [N] ก็จะกลับไปสร้างรูปอักขระหมายเลขเดิมใหม่ ถ้ากด [Y] ก็จะมีเมนูปรากฏให้เลือกตามเมนูคือ กด [S] จะเก็บไฟล์ กด [E] จะกลับไปสร้างอักขระหมายเลขถัดไป ถ้ากดแป้น [Q] ก็จะเป็นการออกจากส่วนการสร้างอักขระ โดยถ้ายังไม่เก็บไฟล์ก็จะถามเรื่องการเก็บไฟล์

การออกจากโปรแกรมอีกทางหนึ่งก็คือ เมื่อค่าตัวแปร ShapeNo=255

ลำดับที่ของอักขระ

เพื่อให้สะดวกในการเลือกรูปอักขระให้ตรงกับข้อความที่เราต้องการจะแสดงบนจอภาพ จึงขอกำหนดลำดับที่ของอักขระทั้งภาษาไทยและภาษาอังกฤษไว้ดังนี้

- ❖ ภาษาอังกฤษเก็บตั้งแต่ลำดับที่ 1 ถึง 94
- ❖ ภาษาไทยเริ่มตั้งแต่ลำดับที่ 96 ไปจนถึงลำดับที่ 163
- ❖ ลำดับที่เหลือคือหมายเลข 164 ไปจนถึงหมายเลข 255 มีไว้ให้ผู้อ่านกำหนดอักขระเอาเองตามความต้องการ

โดยลำดับที่ในหมายเลข 1 และ 2 จะเป็นดังตารางที่ 7.2

ตารางที่ 7.2 *แสดงตารางกำหนดลำดับที่ของอักขระ*

	0	1	2	3	4	5	6	7	8	9
0		!	"	#	$	%	&	,	(	)
1	*	+	,	-		/	0	1	2	3
2	4	5	6	7	8	9	:	;	<	=
3	>	?	@	A	B	C	D	E	F	G
4	H	I	J	K	L	M	N	O	P	Q
5	R	S	T	U	V	W	X	Y	Z	[
6	\	]	^	_	`	a	b	c	d	e
7	f	g	h	i	j	k	l	m	n	o
8	p	q	r	s	t	u	v	w	x	y
9	z	{	¦	}	~	ก	ข	ค	ฆ	
10	ง	จ	ฉ	ช	ซ	ฌ	ญ	ฎ	ฏ	ฐ
11	ฑ	ฒ	ณ	ด	ต	ถ	ท	ธ	น	บ
12	ป	ผ	ฝ	พ	ฟ	ภ	ม	ย	ร	ฤ
13	ล	ว	ศ	ษ	ส	ห	ฬ	อ	ฮ	ะ
14		า	ำ	เ	แ	โ	ใ	ไ	ๆ	ฯ
15	่	๋	ิ	ี	ึ	ื	ุ	ฺ	็	่
16	้	๊	+	์						

ขอให้ผู้อ่านใช้เวลาในช่วงนี้ ทำการสร้างอักขระตามตารางนี้เก็บไว้ล่วงหน้าก่อนที่จะศึกษาในบทที่ 8 ต่อไป โดยต้องยึดหลักดังนี้

❖ แสดงแบบอักขระปกติ ให้สร้างอักขระนั้นชิดด้านซ้ายมือ คือติดกับเลข กล่าวคือไม่สร้างให้อยู่ที่กึ่งกลาง เพราะการแสดงผลในบทที่ 7 นั้น จะยึดขอบขวาสุดที่วาดรูปของแต่ละตัวอักษรเป็นตำแหน่งเริ่มต้นของอักขระตัวถัดไป

❖ การแสดงส่วนล่าง (สระ ุ ู) และส่วนบน (สระ ิ ี ฯลฯ) นั้น ให้สร้างอักขระนั้น ๆ ชิดด้านขวามือ คือด้านเลข 2

❖ การสร้างอักขระส่วนที่เหลือก็ยึดหลักของลำดับที่ 1 หรือ 2 ตามความต้องการของท่านเอง

8 การแสดงอักขระบนจอภาพ

หวังว่าคงได้ใช้เวลาสร้างอักขระกันจนครบถ้วนแล้ว โดยอย่างน้อยคงสร้างได้คนละหนึ่งชุด และคงอยากจะทราบแล้วว่าจากฟอนต์ที่สร้างเก็บไว้นี้เราจะนำมาแสดงตัวอักขระได้สักกี่แบบ ซึ่งเมื่ออ่านบทนี้จบลงก็คงจะพบกับความแปลกใจที่ว่าตัวอักขระเพียงแบบเดียวนี้สามารถใช้เทคนิคนำมาแสดงได้อีกหลายรูปแบบ

มาเริ่มศึกษากันเลยดีกว่า

การกำหนดตัวแปรใช้งาน

เพื่อที่เราจะสามารถเรียกใช้ฟอนต์ของตัวอักขระได้หลายฟอนต์ จึงจำเป็นต้องกำหนดชนิดของข้อมูลของอักขระใหม่ โดยเปลี่ยนมาเป็นการใช้ตัวแปรแบบไดนามิกส์ที่อาศัยหน่วยความจำที่เรียกว่า ฮีป มาช่วยงานแทนที่เคยกำหนดไว้เดิมในบทที่ 6 ดังนั้นจึงขอเสนอการกำหนดตัวแปรใหม่ทั้งหมด ดังรูปที่ 8.1

ซึ่งผลของการเปลี่ยนชนิดของตัวแปรนี้จึงต้องเปลี่ยนชุดคำสั่ง loadfile; มาเป็นของใหม่ตามรูปที่ 8.1 ด้วย นอกจากนี้ยังมีชุดคำสั่งอื่น ๆ ที่อยู่ในบทที่ 7 ให้ใช้อีก

นอกจากนี้ยังมีตัวแปรบางตัวได้กำหนดไว้ล่วงหน้าสำหรับไว้ใช้ในบทที่ 9-10 ด้วย ตัวแปรที่เปลี่ยนมาใช้ข้อมูลชนิดพอยน์เตอร์คือ ShapeHead และ ShapePoly และที่กำหนดใหม่คือ YScope ซึ่งจะนำไปใช้ในกรณีที่ต้องการให้แสดงอักขระในแนวโค้งเว้าหรือโค้งนูน ข้อมูลประเภทนี้จะต้องมีการจองหน่วยความจำฮีปเพื่อรองรับด้วยคำสั่ง new เสมอ

ขอให้พิจารณารูปที่ 8.1 กันก่อน

```
PROGRAM Show_ShapeTable_THAI_BANNER;
uses crt,dos,graph;
type  PolygonType=array[1..255] of pointtype;
      PolyByte=array[1..2] of shortInt;
      PolyInt=array[1..2] of integer;
      Shape_Head=^ShapeH;
      ShapeH=array[1..255] of Integer;
      Shape_Poly=^Shape;
      Shape=array[1..10000] of PolyByte;
      Y_Scope=^Scope;
      Scope=array[0..4000] of integer;
      Data_Banner=record
                    x,y:integer;  str:string;
                    size,patt:byte;
                  end;
const patternMe:array[0..9] of FillpatternType=
      (($FE,$EF,$FE,$EF,$FE,$EF,$FE,$EF),
      ($18,$3C,$7E,$FF,$FF,$7E,$3C,$18),
      ($77,$77,$77,$77,$77,$77,$77,$77),
      ($95,$59,$95,$59,$95,$59,$95,$59),
```

```
($E7,$7E,$E7,$7E,$E7,$7E,$E7,$7E),
($6D,$D6,$6D,$D6,$6D,$D6,$6D,$D6),
($B3,$3B,$B3,$3B,$B3,$3B,$B3,$3B),
($5C,$FF,$C3,$C3,$C3,$C3,$FF,$5C),
($5C,$E7,$7E,$C5,$E7,$7E,$C5,$E7),
($18,$3C,$7E,$FF,$7E,$5A,$18,$18));
var
  SizeOfShape,xStart,Ypoint,PageNo:word;
  ColorBox,ColorBg,PattNo,ShapeNo,TableNo,
  font,High,Enlarge,MaxFont,PlayMode,NoSTR:byte;
  PolyPoint:array[1..255] of PolyInt;
  ShapeHead:array[1..4] of Shape_head;
  ShapePoly:array[1..4] of Shape_Poly;
  Funckey,ch:char;
  Monoscreen:boolean;     Regs:registers;
  XRight,Position:Integer;
  Xpoint:array[1..10] of Integer;
  up,down,Italic,Shadow,Symbol:boolean;
  Ywave:array[0..719] of integer;
  YScope:array[1..10] of Y_Scope;
  BannerData:array[1..10] of Data_Banner;
PROCEDURE LoadFile;
var f:file; PointStart:word;  STT:string;
begin
  MaxFont:=3;
 for font:=1 to Maxfont do
  begin
   mark(ShapeHead[font]); new(ShapeHead[font]);
   mark(ShapePoly[font]); new(ShapePoly[font]);
```

```
    str(font,STT);
    assign(f,'banner'+STT+'.fon'); reset(f,1);
    blockread(f,ShapeNo,1);
    blockread(f,PointStart,2);
    blockread(f,SizeOfShape,2);
    blockread(f,ShapeHead[font]^,510);
    blockread(f,ShapePoly[font]^,SizeOfShape);
    close(f);
  end;  font:=1;
end;
```

รูปที่ 8.1 *ตัวแปรใช้งานและคำสั่ง loadfile ที่ปรับปรุงใหม่*

ตัวแปรที่กำหนดไว้ที่สำคัญมีดังนี้

- xStart นำไปกำหนดการแสดงข้อความแบบเฉียงขึ้นและเฉียงลง
- yPoint นำไปตรวจสอบความกว้างของตัวอักขระแต่ละตัวเพื่อที่จะได้แสดงอักขระตัวต่อไปในจุดที่เหมาะสม ทำให้ข้อความที่ได้มีการปรับระยะห่างตามอักขระแต่ละตัวได้
- font ตัวแปรกำหนดฟอนต์ตัวอักขระที่จะนำมาแสดงบนจอภาพ กำหนดไว้สูงสุด 4 ฟอนต์ กำหนดค่าสูงสุดของจำนวนฟอนต์ด้วยตัวแปร MaxFont แต่ในโปรแกรมกำหนดไว้เพียง 2 ฟอนต์เท่านั้น
- High, Enlarge ตัวแปรกำหนดการแสดงตัวสูง 2 เท่า และตัวกว้าง 2 เท่า
- PlayMode เป็นตัวแปรที่กำหนดรูปแบบการแสดงข้อความ ซึ่งมีค่า 0-5 ดังนี้

0 - แสดงข้อความตามแนวนอนตามปกติ

1 - แสดงข้อความแบบการเคลื่อนที่ของคลื่น ใช้ ^PW(#23) กำหนด

2 - แสดงข้อความแบบโค้งนูนขึ้น ใช้ ^PC(#3) กำหนด

3 - แสดงข้อความแบบโค้งเว้าลง ใช้ ^PV(#22) กำหนด

4 - แสดงข้อความแบบเฉียงลง ใช้ ^PD(#4) กำหนด

5 - แสดงข้อความแบบเฉียงขึ้น ใช้ ^PU(#21) กำหนด

- XPoint, YScope กำหนดไว้สำหรับนำไปหาระยะที่เหมาะสมในการแสดงข้อความแบบโค้งนูนหรือโค้งเว้า
- Ywave กำหนดไว้รองรับระยะที่คำนวณได้ของการแสดงข้อความแบบคลื่น
- Italic, Shadow กำหนดไว้สำหรับการแสดงข้อความแบบตัวเอนและตัวนูน
- bannerData กำหนดไว้สำหรับเก็บรายละเอียดต่าง ๆ ของข้อความที่จะนำมาแสดง
- symbol กำหนดไว้เพื่อรองรับการแสดงอักขระตั้งแต่ลำดับที่ 164 เป็นต้นไป ซึ่งควบคุมด้วยรหัส ^PP (รายการนี้ไม่ได้แสดงการทำงานโดยขอทิ้งไว้ให้เพื่อพัฒนาต่อเอง)

ชุดคำสั่ง loadfile; มีส่วนเปลี่ยนแปลงที่ควรรู้ดังนี้

- กำหนดชื่อไฟล์ของอักขระในชื่อ banner1.fon - banner4.fon จึงให้ก๊อปปี้ไฟล์มาใส่ในชื่อที่กำหนดนี้ ในตัวอย่างกำหนดให้เรียกใช้เพียง 2 ฟอนต์ (MaxFont=2) ถ้ามีฟอนต์มากกว่านี้ก็ให้กำหนด MaxFont ใหม่
- ทำการอ่านไฟล์ข้อมูลเข้ามาโดยต้องมีการกำหนดตำแหน่งของหน่วยความจำฮีปด้วยคำสั่ง mark, new ด้วย และให้ตัวแปร ShapeHead, ShapePoly มาด้วย

❖ กำหนดชื่อไฟล์ด้วยการนำเอาตัวเลข (ในรูปของอักขระ) มาต่อท้ายชื่อไฟล์ banner แล้วตามด้วยสกุล .fon เพื่อเปิดไฟล์ข้อมูลและอ่านข้อมูลทั้งหมดเข้ามา

ขนาดของตัวแปร SizeofShape มีค่าเท่ากับ (ขนาดของไฟล์-515)/2 ซึ่งถ้าตรวจดูทุกฟอนต์ที่ใช้ก็สามารถนำไปกำหนดขนาดของมิติของตัวแปร Shape (ในบรรทัดที่ 9-10) ให้เหมาะสมได้

การคำนวณหาค่าตัวแปร

เพื่อให้การใช้งานในการแสดงข้อความมีรูปแบบที่แตกต่างกัน เราจึงต้องเขียนชุดคำสั่งเพื่อการเก็บค่าบางอย่างไว้ใช้งานตามรูปแบบที่เราต้องการไว้ด้วย ซึ่งตามตัวอย่างนี้ เราจะต้องคำนวณหาค่าของจุดที่จะไปรองรับการแสดงผล 3 แบบคือ การแสดงข้อความแบบการเคลื่อนที่ของคลื่นและการแสดงข้อความแบบโค้งนูนและโค้งเว้า

***การแสดงข้อความให้เป็นแบบคลื่น** (sine wave)* ใช้หลักการคำนวณดังนี้

❖ กำหนดให้แสดงรูปคลื่น 2 ลูกต่อหนึ่งหน้าจอภาพ (ค่าตามแกนนอนจาก 0-GetMaxX) นั่นคือต้องสร้างรูปคลื่นให้เปลี่ยนเฟสไป 720 องศา หรือ 4 เรเดียนในหนึ่งหน้าจอ

❖ ระยะที่เราจะให้เปลี่ยนไปก็คือค่าในแกนยืน (แกน y) ดังนั้นจึงกำหนดตัวแปร Ywave มารองรับค่าที่คำนวณได้ โดยจะคำนวนหาค่า Ywave นี้ในทุก ๆ ค่าของ x (แกน x) เพื่อนำไปรวมกับตำแหน่งต่าง ๆ ในอักขระแต่ละตัวตามแกนยืน

❖ วาดรูปตัวอักษรในตำแหน่งที่คำนวณได้ก็จะได้ข้อความแสดงแบบคลื่นตามต้องการ

การคำนวณหาค่า Ywave นี้เป็นไปตามชุดคำสั่ง WavePoint ในรูปที่ 8.2

การแสดงข้อความแบบโค้งนูนและโค้งเว้า มีหลักการดังนี้

❖ นำข้อความนั้นมาคำนวณหาค่าระยะตามแกน x ประมาณกี่จุด (เพื่อจะได้ความโค้งที่เหมาะสมกับข้อความ) ค่านี้เป็นค่าประมาณเพราะว่าตัวอักขระแต่ละตัวนั้นจะมีความกว้างไม่เท่ากัน การคำนวณนี้จะต้องคิดขนาดของตัวขยายด้วยเพราะตัวขยายจะกินความกว้างเพิ่มขึ้นถึงสองเท่า

❖ คำนวณหาค่าความโค้ง (ความสูงตามแกนยืน) โดยหารัศมีของวงรีครึ่งวงกลมนำมาเก็บในตัวแปร YScope ที่กำหนดให้เป็นตัวแปรสองมิติ เพราะอาจมีข้อความที่ต้องการแสดงแบบนี้มากกว่าหนึ่งข้อความ ซึ่งจะต้องมีความยาวที่ไม่เท่ากัน

การคำนวณค่า YScope อยู่ในชุดคำสั่ง ScopePoint ในรูปที่ 8.2

```
PROCEDURE ScopePoint(content:string; size:byte);
var k,XMax:word; stt:string;

  FUNCTION FindLength(Content:string):word;
    var k,count:word;  Ccount:byte;
    begin  count:=0;  Ccount:=1;
       for k:=1 to length(Content) do
        begin
          if content[k]=#5 then Inc(Ccount);
          if Ccount>2 then Ccount:=1;
          if content[k] in [#32..#214] then
                           inc(count,Ccount);
        end;
       FindLength:=Count*Size*20;
    end;
```

```
begin
   XMax:=FindLength(Content);
   for k:=0 to XMax+50 do
      Yscope[NoSTR]^[k]:=round(50*sin((k/XMax)*pi));
end;

PROCEDURE WavePoint;
var k:word;
      ShapeSTR:string;
begin
      for k:=0 to GetMaxx d
      YWave[k]:=round(24*sin((k/getmaxx)*4*pi));
end;
```

รูปที่ 8.2 ชุดคำสั่งเพื่อการคำนวณหาจุดต่าง ๆ เพื่อแสดงข้อความแบบโค้งและแบบคลื่น

การแสดงอักขระบนจอภาพ

การนำอักขระมาแสดงบนจอภาพให้ตรงกับข้อความที่เรากำหนดมาให้นั้น ผู้เขียนขอใช้ภาษาไทยรหัสเกษตรเป็นต้นแบบ (เพราะผู้เขียนมีเฉพาะการ์ดรหัสนี้เท่านั้น) โดยได้กำหนดอักขระที่สร้างไว้แล้วในบทที่ 6 (ตารางที่ 6.2)

ลองมาดูภาษาอังกฤษ เช่น อักขระ A จะอยู่ลำดับที่ 33 นั่นคือหาได้จากรหัสแอสกี-32 จึงได้ลำดับของ A เป็น 65-32=33 เป็นต้น ตัวอักขระอื่น ๆ ก็คิดแบบเดียวกัน นั่นคือถ้ารหัสแอสกีนั้นเป็นภาษาอังกฤษก็หาลำดับในไฟล์ข้อมูล (ในโปรแกรมใช้ชื่อตัวแปรว่า TableNo) ด้วยการลบรหัสแอสกีออกด้วยเลข 32

สังเกตภาษาไทยบ้างจะพบว่าอักขระ ก อยู่ที่ลำดับ 96 ดังนั้นภาษาไทยจึงหาตัวเลขที่เอามาลบออกจาก 161-96=65 (161 คือรหัสแอสกีของ ก)

หากต้องการจะใช้อักขระลำดับที่ 164 เป็นต้นไป ก็อาจทำได้โดยการใช้รหัสควบคุม เช่น กำหนดให้ ^PP เข้า-ออกจากอักขระกลุ่มนี้โดยใช้คู่กับอักขระภาษาอังกฤษ ดังนั้นค่าของลำดับที่จึงบวกเพิ่มจากรหัสแอสกีอีก 164-1=163 (อักขระ! คือข้อมูลลำดับที่ 1 ของไฟล์ตัวอักษร) ทั้งนี้ท่านจะต้องมีการสร้างอักขระไว้ด้วย

เมื่อสามารถหาลำดับที่ของตัวอักขระได้ เราจะรู้จุดเริ่มต้นของข้อมูลได้จากตัวแปร ShapeHead ตามลำดับที่คำนวณได้ และสิ้นสุดเมื่อค่าตัวแปร ShapePoly ที่ตรวจพบมีค่าเป็น 86 (เก็บไว้ตอนสร้างอักขระจากบทที่ 6)

ทำให้เราสามารถนำข้อมูลของอักขระแต่ละตัวนั้น ๆ มาแสดงบนจอภาพได้อย่างถูกต้อง และเนื่องจากเราได้กำหนดให้การเก็บค่าของข้อมูลเป็นการอ้างอิงกับคู่ลำดับที่ผ่านมาแบบเวกเตอร์ (มีทั้งค่าบวกและลบ) จึงสามารถทำให้เปลี่ยนแปลงขนาดของอักขระได้โดยที่ยังมีความละเอียดของตัวอักขระเหมือนเดิม พร้อมทั้งยังสามารถทำให้เป็นอักขระที่มีรูปร่างต่าง ๆ กัน เช่น ตัวขยาย 2 เท่า ตัวสูง 2 เท่า หรืออื่น ๆ ได้

ชุดคำสั่งที่ใช้ในการแสดงอักขระแต่ละตัวจะอยู่ในชุดคำสั่ง ShowPoly ในรูปที่ 8.3 ขอให้ท่านทำความเข้าใจด้วยตนเอง

```
PROCEDURE   ShowPoly(Xx,Yy:integer;  size:byte);
var
    Poly:PolygonType;  P,NO,Patt:byte;
    x,y,Xold,YOld,Ystart,XItalic,Udpix:Integer;
begin
    Ystart:=Yy;  UdPix:=0;
```

```
        if up then Udpix:=-Size*15;
        if Down then Udpix:=Size*13;
        if TableNO in [156,160] then inc(xx,6*size);
        Xx:=Xx-Size*PolyPoint[1][1];
        Yy:=Yy-Size*PolyPoint[1][2];
        if PolyPoint[1][1]=0 then XRight:=Xx;
        Xold:=Xx; YOld:=Yy; No:=2;     SetPattern;
    Repeat P:=1;
        Repeat
            x:=Xx+Size*PolyPoint[No][1]*Enlarge;
            y:=Yy-Size*PolyPoint[No][2]*high;
            Poly[p].x:=x;   Poly[p].y:=y+Udpix;
            if (PolyPoint[1][1]=0) and (XRight<x) and
                      ((YPoint-y)<size*18*high) then  XRight:=x;
            if Italic then Poly[p].x:=x+round((Ystart-y) div 4);
            if PlayMode=5 then Poly[p].y:=Poly[p].y-
                          round((x+(PageNo-1)*GetMaxX-Xstart)/12);
            if PlayMode=4 then Poly[p].y:=Poly[p].y+
                          round((x+(PageNo-1)*GetMaxX-Xstart)/12);
            if (PlayMode=1) and (x>=0) then Poly[p].y:=Poly[p].y+
                                        Ywave[x mod (getMaxX+1)];
            if (PlayMode=2) and (x>=-Size*25) then
                Poly[p].y:=Poly[p].y-YScope[NoSTR]^[x+(PageNo-1)*
                                        GetMaxX-Xpoint[NoSTR]+20];
            if (PlayMode=3) and (x>=-Size*25) then
                Poly[p].y:=Poly[p].y+YScope[NoSTR]^[x+(PageNo-1)*
                                        GetMaxX-Xpoint[NoSTR]+20];
            Inc(NO); Inc(p); Xx:=x; Yy:=y;
        until PolyPoint[No][1] in [86,99];
```

```
if xx>=-Size*25 then FillPoly(P-1, Poly);
  if PolyPoint[No][1]=99 then
  begin
     Xx:=Xold; Yy:=Yold; Inc(No); SetPattern;
     if PolyPoint[No][1]=127 then
     begin
        Patt:=PattNo; PattNo:=0; SetPattern;
        PattNo:=Patt; Inc(No);
     end;
  end;
Until PolyPoint[No][1]=86;
end;
```

รูปที่ 8.3 *ชุดคำสั่งในการแสดงอักขระตามขนาดที่กำหนด*

โปรแกรมหลักการแสดงอักขระ

เราจำเป็นต้องเขียนชุดคำสั่งเพื่อที่จะได้นำข้อความที่เรากำหนดไปแสดงข้อความบนจอภาพ ให้ถูกต้องตรงกับความต้องการของเรา โดยต้องคำนึงถึงสิ่งที่เราได้กำหนดไว้แล้ว เช่น รูปแบบการแสดงข้อความ การเลือกฟอนต์ตัวหนังสือ ฯลฯ นั่นคือ เราต้องกำหนดตัวแปรไว้สำหรับการใช้งานด้วย

ผู้เขียนได้กำหนดตัวแปรและรหัสในการทำงานไว้ดังนี้

- ตัวแปร PlayMode=1 และรหัส ^PW(#23) ใช้ในการแสดงข้อความแบบคลื่น
- ตัวแปร PlayMode=2 และรหัส ^PC(#3) ใช้ในการแสดงข้อความแบบโค้งนูนขึ้น

- ตัวแปร PlayMode=3 และรหัส ^PY(#25) ใช้ในการแสดงข้อความแบบโค้งเว้าลง
- ตัวแปร PlayMode=4 และรหัส ^PD(#4) ใช้ในการแสดงข้อความแบบเฉียงลง
- ตัวแปร PlayMode=5 และรหัส ^PU(#21) ใช้ในการแสดงข้อความแบบเฉียงขึ้น
- ตัวแปร Enlarge และรหัส ^PE(#5) สำหรับการแสดงตัวแบบขยายหรือกว้าง 2 เท่า
- ตัวแปร High และรหัส ^PH(#8) สำหรับตัวสูง 2 เท่า
- ตัวแปร Italic และรหัส ^PL(#12) สำหรับแสดงตัวเอน
- ตัวแปร Shadow และรหัส ^PS(#19) สำหรับแสดงตัวนูน
- ตัวแปร Symbol และรหัส ^PP(#16) สำหรับการแสดงอักขระพิเศษ
- ตัวแปร Up และรหัส ^PT(#20) สำหรับการแสดงแบบตัวยกกำลัง
- ตัวแปร Down และรหัส ^PV(#22) สำหรับการแสดงแบบตัวห้อยท้าย
- ตัวแปร TableNo ใช้สำหรับเก็บลำดับของข้อมูล (เปลี่ยนมาจากรหัสแอสกีของอักขระในข้อความอีกที) ซึ่งทำให้เราสามารถกำหนดเงื่อนไขได้ง่ายขึ้น โดยการดูรายละเอียดจากตารางที่ 6.2
- ตัวแปร no, noback ใช้สำหรับเก็บลำดับที่ของตัวอักขระที่กำลังตรวจสอบและลำดับที่ได้แสดงไปแล้วก่อนหน้านี้ เพื่อมากำหนดการแสดงอักขระ (บางตัว) ให้มีความเหมาะสมยิ่งขึ้น

❖ ตัวแปร font และรหัส ^P\,^P],^P^,^P_ (#28-#31) ใช้เพื่อเลือกฟอนต์ของตัวอักขระ

นำมาเขียนชุดคำสั่งในการทำงานได้ดังรูปที่ 8.4 ขอให้ท่านทำความเข้าใจในแต่ละบรรทัดด้วยตนเองให้เข้าใจก่อน

```
PROCEDURE ShowBanner(x,y:Integer; Content:string;Size,Patt:byte);
var k,No,B,kk,Exit,TableNoOld,noback,ss:byte;
    Shape:word; xpix,Ypix:integer;
Begin
   PattNo:=Patt;          Content:=Content+#255;  Ypoint:=y;
   Shadow:=False;         Italic:=False; up:=false;
   down:=false;           Symbol:=false;
   Enlarge:=1;            High:=1; PlayMode:=0;
   if size<3 then ss:=size else ss:=3;
    k:=pos(#3,Content);
      if k<>0 then
       begin
         delete(Content,k,1); PlayMode:=2
       end;
    k:=pos(#25,Content);
      if k<>0 then
       begin
         delete(Content,k,1); PlayMode:=3
       end;
    if (PlayMode in [2,3]) and (PageNo=1) then
      begin
        ScopePoint(content,size);
```

```
          xpoint[NoSTR]:=x;
        end;
      k:=pos(#21,Content);
        if k<>0 then
        begin
          delete(Content,k,1); PlayMode:=5;
        end;
      k:=pos(#4,Content);
        if k<>0 then
        begin
           delete(Content,k,1); PlayMode:=4;
        end;
      if (PlayMode in [4,5]) and (PageNo=1) then  Xstart:=x;
      k:=pos(#23,Content);
        if k<>0 then
        begin
           delete(Content,k,1); PlayMode:=1;
        end;
      k:=pos(#207,content);
        if k<>0 then
        begin
           if content[k+1] in [#224..#227] then
            begin
              content[k]:=content[k+1];
              content[k+1]:=#206; inc(k);
            end;
           content[k]:=#206; insert(#222,content,k);
        end;
```

```
        k:=1;  Exit:=0;
        setlinestyle(0,123,3);
     if shadow or (Size<4) then setlinestyle(0,123,1);
Repeat
  repeat
     if Content[k] in [#28..#31] then
      begin
         font:=ord(Content[k])-27;
           if font>MaxFont then font:=1;
      end;
     if Content[k]=#5  then
      begin
         inc(Enlarge); if Enlarge=3 then Enlarge:=1;
      end;
      if Content[k]=#12 then
        if Italic then Italic:=false Else Italic:=true;
      if Content[k]=#16 then
        if Symbol then Symbol:=false Else Symbol:=true;
      if Content[k]=#19 then
        if Shadow then Shadow:=false Else Shadow:=true;
      if Content[k]=#20 then
        if Up then Up:=false Else Up:=true;
      if Content[k]=#22 then
        if Down then Down:=false Else Down:=true;
      if Content[k]=#8 then
        begin
          inc(High); if High=3 then High:=1;
        end;
```

```
        if Content[k]<#32 then inc(k);
     until Content[k] in [#32..#255];
  if Content[k] in [#32,#160] then Inc(x,20*Size)
  else
  begin
     no:=Ord(Content[k])-32;
     noback:=Ord(Content[k-1])-32;
  if Content[k]>#160 then
   begin
      no:=Ord(Content[k])-65;
      noback:=Ord(Content[k-1])-65;
   end;
  if Symbol and (content[k] in [#33..#123]) then inc(no,163);
     Shape:=ShapeHead[font]^[No]-1;
     xpix:=0; Ypix:=0; TableNo:=no;
  if (TableNo in [145..147]) and (font in [1,2]) then
     Dec(x,7*Size);
  if TableNo in [159..163] then
   begin
     if noback= 142 then xpix:=-Size*2;
     if (noback in [142,152..157,222]) or
                 (content[k+1]=#222) then
       begin
        inc(Ypix,size*8*high);
        if Italic then inc(x,size*2);
       end;
   end;
```

```
if (TableNo in [100,101,109,116,119,120,123,124,
   128,131,133,135,136]) and (Font in [1..2]) then
      if noback in [133,143,144,145..147] then dec(x,size*4);
if (TableNo in [150,151]) and (Font=2) then
      if noback in [101,109] then dec(x,size*4);
   B:=0;
      Repeat Inc(B);  Inc(Shape);
         PolyPoint[B][1]:=ShapePoly[font]^[Shape][1];
         PolyPoint[B][2]:=ShapePoly[font]^[Shape][2];
      until ShapePoly[font]^[Shape][1]=86;
   if shadow then
      for kk:=ss downto 1 do ShowPoly(x+xpix+kk,y-Ypix+kk,size);
     ShowPoly(x+xpix,y-Ypix,size);
   if PolyPoint[1][1]=0 then
     begin
       TableNoOld:=TableNo; x:=XRight+Size*4;
     end;
 end;
     inc(k);
   if X>GetMaxX then
      begin
        Position:=k-1;
        if Content[Position]>#160 then
             no:=Ord(Content[Position])-65
        Else no:=Ord(Content[Position])-32;
      if Symbol and (content[Position] in [#33..#123]) then inc(no,131);
      if ShapePoly[font]^[ShapeHead[font]^[No]][1]=0 then Inc(Exit);
      if Exit=2 then k:=length(Content);
     end;
```

```
Until k>length(Content)-1;
if Exit=0 then Position:=length(Content)+1;
end;
```

รูปที่ 8.4 *ชุดคำสั่งหลักในการนำข้อความไปแสดงในรูปแบบที่กำหนด*

การทำงานของชุดคำสั่ง ShowBanner มีลักษณะดังนี้

- ❖ ส่งค่า x,y ข้อความ ขนาดของข้อความ และรูปแบบพื้นมาให้
- ❖ ตรวจสอบรหัสรูปแบบการแสดงผล (รหัส ^PC,^PY,^PW,^PD,^PU) เพื่อกำหนดค่า PlayMode
- ❖ ตรวจการเขียน ่า ้า ๊า ๋า เพื่อจะสลับที่ให้ตรงกับการที่จะนำข้อความไปแสดง
- ❖ ตรวจการกำหนดลำดับที่ของฟอนต์ที่จะแสดง
- ❖ ตรวจลักษณะการแสดงอักขระ (รหัส ^PL,^PP,^PS,^PT,^PV)
- ❖ ถ้าตรวจพบว่าเป็นอักขระแอสกี 32,160 ก็ให้ขยับค่า x ไปเท่ากับ 20xsize (เพราะเราไม่ได้สร้างอักขระ 2 ตัวนี้ไว้)
- ❖ ถ้าเป็นรหัสอื่น ๆ ก็ให้แสดงอักขระนั้น ๆ ตามลำดับการทำงานดังนี้

1. ตรวจสอบการหาลำดับที่ของอักขระ เพื่อหาจุดเริ่มต้นของข้อมูลจาก ShapeHead[font]^[no]

2. มีการเปลี่ยนแปลงค่า x ในบางอักขระเพื่อให้ได้การแสดงผลที่มีความสวยงาม ส่วนนี้จะอยู่ที่การกำหนดเอาเอง (โดยไม่จำเป็นต้องตรงกับที่กำหนดนี้ก็ได้)

3. เริ่มนำข้อมูลจากจุดเริ่มต้นมาใส่ในตัวแปร PolyPoint ทีละหนึ่งจุดจนกว่าจะพบค่า 86 ก็จะเรียกชุดคำสั่ง ShowPoly เพื่อนำอักขระ

นั้น ๆ ไปแสดงบนจอภาพ หากมีการกำหนดเป็นตัวนูนก็ให้แสดงตัวนั้นหลาย ๆ ครั้ง ณ ตำแหน่งที่เหมาะสม

4. ถ้าตรวจพบว่าข้อมูลลำดับแรก (ตัวแปร PolyPoint[1][1]) มีค่าเท่ากับ 0 ให้เพิ่มค่า x ขึ้นอีก Sizex5 ซึ่งจะเป็นช่องว่างระหว่างตัวอักขระ ข้อมูลนี้เราสามารถกำหนดไว้ตั้งแต่ตอนสร้างอักขระในบทที่ 7 ซึ่งถ้าเราเลือกตำแหน่งเริ่มต้นเป็น 2 หรือ 3 ก็จะไม่มีการเพิ่มช่องว่างนี้

5. เลื่อนไปตัวอักขระที่อยู่ถัดไปจนกว่าจะหมดข้อความ แต่ถ้าค่า x มากกว่าขอบเขตของจอภาพ ก็จะเป็นการสิ้นสุดการทำงานของชุดคำสั่งนี้

เราก็จะได้แนวทางการนำข้อมูลตัวอักขระไปแสดงบนจอภาพแล้ว ในบทที่ 9 เราจะมาศึกษาวิธีการแสดงข้อความแบบต่าง ๆ กันต่อไป

เทคนิค 9 การแสดงข้อความ

เมื่อเราได้แนวทางในการแสดงข้อความแล้ว ก็ต้องนำมากำหนดรูปแบบในการแสดงข้อความให้มีความหลากหลาย เช่น แสดงข้อความแบบโค้ง แบบนูน หรือแบบคลื่น ฯลฯ ซึ่งถ้าหากมองเพียงผิวเผินจะดูเหมือนว่ายากเกินกว่าที่จะทำได้

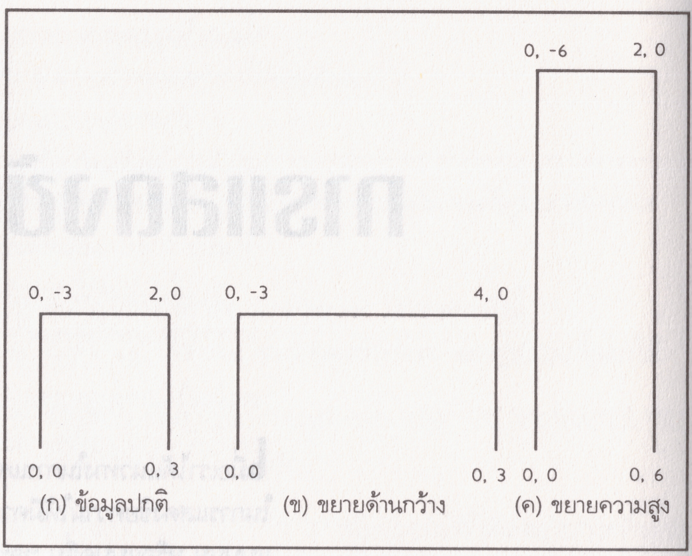
มาดูหลักการที่จะใช้ในการแสดงข้อความกันสัก 2-3 แบบ

โปรแกรมในบทนี้จะใช้งานร่วมกับโปรแกรมในบทที่ 8 ดังนั้นจึงต้องนำชุดคำสั่งต่าง ๆ ในบทที่ 8 มาใช้งานด้วย

การแสดงข้อความแบบขยายและตัวสูง 2 เท่า

เพื่อความเข้าใจในเรื่องการแสดงตัวอักขระในแต่ละรูปแบบนั้น ผู้เขียนขอสมมติข้อมูลของอักขระแบบง่าย ๆ เพียงแค่ 4 คู่ลำดับเพื่อประกอบการอธิบายดังรูปที่ 9.1

คำอธิบายต่อไปนี้อาจมีความเกี่ยวพันกับโปรแกรมในรูปที่ 7.3 และ 7.4 ด้วย จึงขอให้ตรวจสอบจุดที่ได้อธิบายของแต่ละคำสั่งเอาเอง



รูปที่ 9.1 *แสดงข้อมูลแบบปกติ ข้อมูลแบบขยายด้านกว้างและข้อมูลแบบขยายความสูง*

โดยตัวเลขที่เก็บไว้ในไฟล์ข้อมูลจะเป็นดังรูป (ก) ซึ่งจะได้ข้อมูลดังนี้

- ❖ จุดที่ 1 เก็บค่า 0, 0 ซึ่งถือว่าเป็นจุดเริ่มต้น
- ❖ จุดที่ 2 เก็บค่า 0, -3 (ค่าที่เก็บนี้เป็นการเก็บเมื่อเทียบกับจุดที่ 1) หมายถึงว่าค่าในแกน x ไม่เปลี่ยนแปลง แต่ค่าในแกน y จะขยับขึ้นไปข้างบน 3 จุด
- ❖ จุดที่ 3 เก็บค่า 2, 0 (ค่าที่เก็บนี้เป็นการเก็บเทียบกับจุดที่ 2) นั่นคือจุดนี้อยู่ระดับเดียวกันกับจุดที่ 2 แต่จะห่างไปทางขวา 2 จุด
- ❖ จุดที่ 4 เก็บค่า 0, 3 (ค่าที่เก็บนี้เป็นการเก็บเทียบกับจุดที่ 3) ซึ่งแสดงว่าจุดที่ 4 อยู่ด้านล่างของจุดที่ 3 ห่างลงมา 3 จุด

เมื่อจบข้อมูลนี้จะต้องมีการขยับจุดเริ่มต้นไปทางขวาด้วย เพราะข้อมูลของจุดที่ 1 มีค่าเป็น 0, 0

ถ้านำข้อมูลของอักขระนี้มาแสดงบนจอภาพ จะแสดง ณ ตำแหน่งที่คำนวณได้ (สมมติตอนนั้นคู่ลำดับของจุดเริ่มต้นเป็น 200, 100) ตามปกติค่าตัวแปร High = 1 และ Enlarge =1 โดยที่ตัวแปร High จะกระทำกับข้อมูลในแกน y และตัวแปร Enlarge จะกระทำกับข้อมูลในแกน x

- ❖ จุดที่ 1 เป็น 200+0X1,100+0X1 = 200+0,100+0 = 200,100
- ❖ จุดที่ 2 เป็น 200+0X1,100-3X1 = 200+0,100-3 = 200,97
- ❖ จุดที่ 3 เป็น 200+2X1,97+0X1 = 200+2,97+0 = 202,97
- ❖ จุดที่ 4 เป็น 202+0X1,97+3X1 = 202+0,97+3 = 202,100

เมื่อกำหนดเป็นตัวขยายจะได้ข้อมูลแบบรูป (ข) เพราะค่าตัวแปร High = 1 และ Enlarge = 2 จะได้จุดตามลำดับในการแสดงบนจอภาพดังนี้

- ❖ จุดที่ 1 เป็น 200+0X2,100+0X1 = 200+0,100+0 = 200,100
- ❖ จุดที่ 2 เป็น 200+0X2,100-3X1 = 200+0,100-3 = 200,97
- ❖ จุดที่ 3 เป็น 200+2X2,97+0X1 = 200+4,97+0 = 204,97
- ❖ จุดที่ 4 เป็น 204+0X2,97+3X1 = 204+0,97+3 = 204,100

เมื่อกำหนดให้เป็นตัวสูง 2 เท่า จะได้ข้อมูลแบบรูป (ค) เพราะค่าตัวแปร High = 2 และ Enlarge = 1 จะได้จุดตามลำดับในการแสดงบนจอภาพโดยจะมีค่าดังนี้

- ❖ จุดที่ 1 เป็น 200+0X1,100+0X2 = 200+0,100+0 = 200,100
- ❖ จุดที่ 2 เป็น 200+0X1,100-3X2 = 200+0,100-6 = 200,94
- ❖ จุดที่ 3 เป็น 200+2X1,94+0X2 = 200+2,94+0 = 202,94
- ❖ จุดที่ 4 เป็น 202+0X1,94+3X2 = 202+0,94+6 = 202,100

เพียงเราใช้แนวคิดนี้ไปประกอบกับคำสั่งในโปรแกรม ก็จะสามารถแสดงข้อความเป็นแบบปกติ ตัวขยายหรือตัวสูงในป้ายประกาศได้

การแสดงตัวเอนและตัวแบบโปร่ง

ในการทำตัวเอนนั้นใช้วิธีกำหนดให้ค่าคู่ลำดับในแกน x มีค่าเปลี่ยนไปดังนี้

❖ กำหนดจุดเริ่มต้นตามแกน y (ตามตัวอย่างคือ y = 100) ให้จำค่านี้ไว้ในตัวแปรชื่อ Ystart

❖ ถ้าจุดในแกน y มีค่าไม่เท่ากับจุดเริ่มต้น ก็จะมีการเปลี่ยนแปลงค่าตามแกน x คือ

1. ถ้าน้อยกว่าจุดเริ่มต้นก็เพิ่มค่าในแกน x
2. ถ้ามากกว่าจุดเริ่มต้นก็ลดค่าในแกน x
   (ตรวจจากตัวแปร Ystart และ y)

ค่าที่มีการเปลี่ยนแปลงนี้ ตามโปรแกรมในรูปที่ 7.3 จะกำหนดค่าที่เลื่อนจาก (Ystart-y) div 4 ถ้าต้องการให้เอียงมากหรือน้อยกว่านี้ก็ให้เปลี่ยน-แปลงค่าตัวเลข 4 ที่เอาไปหาร ผลคือ ถ้าค่า y น้อยกว่า Ystart จะได้ค่าที่คำนวณนี้เป็น + จึงทำให้ได้ค่าตามแกน x เพิ่มขึ้น ถ้าค่า y มากกว่า Ystart ก็จะได้ค่าเป็น - ค่าตามแกน x จึงลดลง (กรณีของการแสดงสระ ุ ู หรืออักขระ ญ ฐ ฎ ฏ เป็นต้น)

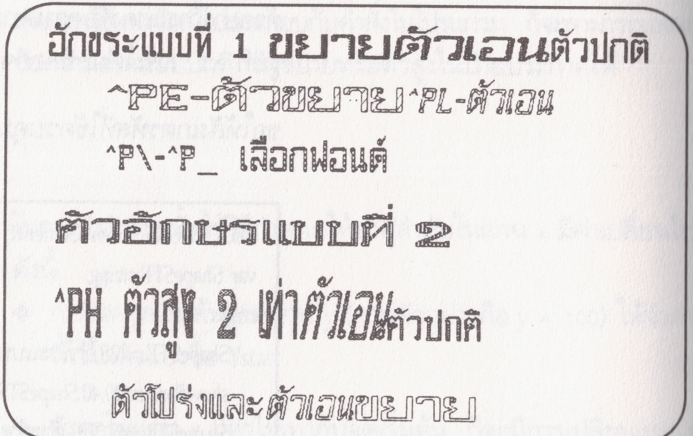
ส่วนตัวแบบโปร่งทำได้ง่ายด้วยการกำหนดรูปแบบพื้นเป็น 0 เท่านั้น

ตัวอย่างการแสดงข้อความตามที่กล่าวมาทั้งหมด ให้ดูตัวโปรแกรมจากรูปที่ 9.2 และได้ผลที่ลอกจากจอภาพดังรูปที่ 9.3

ขอให้สังเกตรหัสที่ใช้ควบคุมการแสดงผลในโปรแกรมด้วย

```
PROCEDURE ShowContent1;
var ShapeSTR:string;
begin
   ShapeSTR:=#28'อักขระแบบที่ 1 '#5'ขยาย'#12'ตัวเอน'#5#12'ตัวปกติ';
   showBanner(20,40,ShapeSTR,1,1);
   ShapeSTR:=#5'^PE-ตัวขยาย'#5#12'^PL-ตัวเอน ';
   showBanner(80,80,ShapeSTR,1,13);
   ShapeSTR:='^P\-^P_ เลือกฟอนต์';
   showBanner(80,125,ShapeSTR,1,12);
   ShapeSTR:=#29#5'ตัวอักษรแบบที่ 2 ';
   showBanner(30,180,ShapeSTR,1,7);
   ShapeSTR:=#8'^PH ตัวสูง 2 เท่า'#12'ตัวเอน'#12#8'ตัวปกติ';
   showBanner(30,250,ShapeSTR,1,19);
   ShapeSTR:=#28'ตัวโปร่งและ'#12'ตัวเอน'#12#5'ขยาย';
   showBanner(90,310,ShapeSTR,1,0);
   readchr;
end;
```

รูปที่ 9.2 *ชุดคำสั่งเพื่อนำทดสอบการแสดงข้อความบนจอภาพ*



รูปที่ 9.3 *ภาพที่ลอกจากจอภาพเมื่อรันชุดคำสั่งในรูปที่ 9.1*

การแสดงข้อความแบบเฉียงขึ้นลง

เนื่องจากการเก็บข้อมูลของแต่ละอักขระมักจะมีรูปแบบดังที่กล่าวมาแล้ว ดังนั้นเราจึงสามารถนำมาแสดงบนจอภาพได้หลาย ๆ รูปแบบ เช่น การแสดงตัวอักขระแบบเฉียงขึ้นและแบบเฉียงลง ฯลฯ ทั้งนี้เพราะใช้หลักที่ว่าตัวอักขระทุกตัวมีมุมล่างซ้าย ณ ตำแหน่งเดียวกัน (จุดนี้ไม่จำเป็นว่าจะต้องลงจุดเสมอไป) เพียงแต่จะมีจุดเริ่มต้นที่ไม่ใช่จุดเดียวกัน ในที่นี้หมายความว่าอักขระทุกตัวจะมีข้อมูลลำดับที่ 1 อ้างอิงจากจุดเดียวกัน เพียงแต่จุดเริ่มต้น (คู่ลำดับที่ 2) จะเป็นไปตามข้อมูลตอนสร้างขึ้นมา โดยขอให้สังเกตคำสั่งในรูปที่ 7.3 ในชุดคำสั่ง ShowPoly ที่หาค่าเริ่มต้นดังนี้

```
Xx:=Xx-Size*PolyPoint[1][1];
Yy:=Yy-Size*PolyPoint[1][2];
```

ดังนั้นเราจึงนำมาแสดงข้อความตามแนวเฉียงได้ดังนี้

❖ กำหนดจุดเริ่มต้นของข้อความตามแนวราบ (แกน x) ในตัวแปรชื่อ Xstart เพื่อนำไปกำหนดการแสดงในแนวเฉียงขึ้นหรือลง

❖ เมื่อตรวจพบรหัส ^PD(#4) จะทำให้ตัวแปร PlayMode = 4 หรือพบ ^PU(#21) ก็จะได้ตัวแปร PlayMode = 5 (กำหนดพร้อมกันสองรหัสจะได้ตัวแปร PlayMode = 4 เพราะตรวจรหัส ^PD ทีหลัง)

❖ เมื่อนำอักขระนี้ไปแสดงผลด้วยชุดคำสั่ง ShowPoly ก็จะทำการบวกค่าของคู่ลำดับในแกน y ให้เป็นค่าใหม่ (ค่าในแกน x คงเดิม) ตามคำสั่ง

Poly[p].y:=Poly[p].y+round((x-Xstart)/12); ในกรณีที่แสดงในแนวเฉียงลง

Poly[p].y:=Poly[p].y-round((x-Xstart)/12); ในกรณีที่แสดงในแนวเฉียงขึ้น

หากต้องการให้แสดงในแนวเฉียงขึ้นหรือเฉียงลงมีความเอียงแตกต่างจากนี้ก็ให้เปลี่ยนตัวเลข 12 ไปตามความต้องการได้ นั่นคือ อาจกำหนดให้แสดงในแนวเฉียงได้หลาย ๆ แนว

การกำหนดตัวแปร size ก็จะทำให้ได้ขนาดของอักขระที่มีขนาดต่าง ๆ กันออกไป ตัวอย่างการแสดงผลทั้งสามแบบนี้เป็นดังโปรแกรมในรูปที่ 9.4 และได้ภาพที่ลอกจากหน้าจอดังรูปที่ 9.5

```
PROCEDURE ShowContent2;
var ShapeSTR:string;
begin
  clsGR(0,0,GetMaxX,GetMaxY);
  ShapeSTR:=#28#4'ความลาดชันคือ';
  showBanner(80,30,ShapeSTR,1,11);
  ShapeSTR:=#29#21'แนวที่สูงต่ำไม่เท่ากัน';
  showBanner(80,110,ShapeSTR,1,6);
  ShapeSTR:=#29#19'^PS-ตัวนูน';
  showBanner(30,200,ShapeSTR,2,20);
  ShapeSTR:=#29#19#12'ตัวนูน';
  showBanner(320,200,ShapeSTR,4,0);
  ShapeSTR:=#21'^PU-ข้อความเฉียงขึ้น';
  showBanner(90,280,ShapeSTR,1,15);
  ShapeSTR:=#4'^PD-ข้อความเฉียงลง';
  showBanner(220,300,ShapeSTR,1,18);
  readchr;
end;
```

รูปที่ 9.4 *ชุดคำสั่งนำข้อความไปแสดงผลบนจอภาพแบบที่ 2*

รูปที่ 9.5 *ภาพที่ลอกจากจอภาพของชุดคำสั่งในรูปที่ 9.4*

การแสดงข้อความแบบคลื่นและแนวโค้ง

ในแนวทางเดียวกันเราสามารถเอาข้อความมาแสดงในแนวโค้งได้ ในตัวอย่างที่ให้มานี้จะมีอยู่เพียง 3 แบบคือ แบบโค้งนูนขึ้น โค้งเว้าลง และแสดงแบบคลื่น โดยอาศัยหลักการดังนี้

- ❖ ตรวจสอบรหัสการแสดงผลคือ ^PC(#3) จะเป็นแบบโค้งนูน ^PY(#25) จะเป็นแบบโค้งเว้า และ ^PW(#23) จะเป็นแบบคลื่น
- ❖ คำนวณหาแนวของคลื่นไว้ล่วงหน้า ตามที่อธิบายมาแล้วในชุดคำสั่ง WavePoint (บทที่ 7) ส่วนในการแสดงแบบโค้งนั้นจะต้องคำนวณตามข้อความที่แสดง เพราะมีความประสงค์จะให้ได้ความโค้งที่พอดีกันทั้งเริ่มต้นและจบข้อความ จึงต้องส่งข้อความนั้น ๆ ไปคำนวณหาค่า YScope ตามชุดคำสั่ง ScopePoint

สำหรับค่าของ YScope ตามที่แสดงในตัวอย่างนี้อาจจะยังไม่ถูกต้องนัก ทั้งนี้เพราะขนาดความกว้างของแต่ละอักขระไม่เท่ากัน แต่การคำนวณจะใช้ความกว้างขนาด 20*size เท่ากันหมด ดังนั้นหากข้อความมีความโค้งไม่เหมาะสมก็ให้แก้ปัญหาด้วยการเพิ่มช่องว่างหน้าหรือหลังข้อความเอาเอง

- ❖ กรณีที่เป็นการแสดงแบบโค้งต้องจดจำตำแหน่งเริ่มต้นของข้อความไว้ในตัวแปร Xpoint[NoSTR]; ทุกครั้ง
- ❖ เมื่อมีการแสดงอักขระ เราจะเปลี่ยนแปลงข้อมูลในแกน y โดยนำค่าของตัวแปรที่คำนวณได้ในลำดับแรกมารวมกับค่าของแกน y เดิมตามคำสั่ง

Poly[p].y:=Poly[p].y + YWave[x mod (getmaxx+1)]; (แสดงแบบคลื่น)

Poly[p].y:=Poly[p].y - YScope[NoSTR]^[x-Xpoint[NoSTR]+20]; (แสดงแบบโค้งนูน)

Poly[p].y:=Poly[p].y + YScope[NoSTR]^[x-Xpoint[NoSTR]+20]; (แสดงแบบโค้งเว้า)

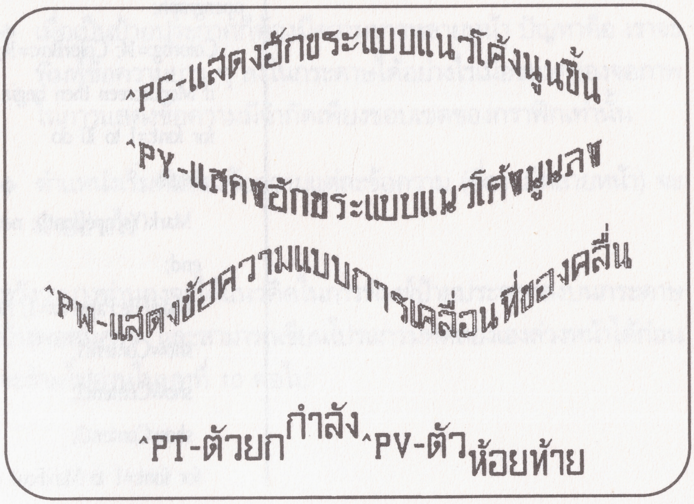
ซึ่งจะทำให้ได้ข้อความที่แสดงตามแบบที่ต้องการได้

ในแต่ละกรณีนี้เรายังสามารถกำหนดให้เป็นตัวเอน ตัวนูน ตัวขยาย หรือตัวสูงได้อีกด้วย จึงทำให้รูปแบบของอักขระที่เก็บไว้เพียงแบบเดียวนี้สามารถแสดงผลบนจอภาพได้มากแบบขึ้น

ตัวอย่างในรูปที่ 9.6 เป็นการแสดงข้อความแบบโค้งนูน โค้งเว้า และแบบคลื่นโดยได้ลอกภาพหน้าจอมาให้ดูเป็นตัวอย่างตามรูปที่ 9.6

```
PROCEDURE ShowContent3;
var ShapeSTR:string;
begin
    clsGR(0,0,GetMaxX,GetMaxY);
    ShapeSTR:=#28#3'^PC-แสดงอักขระแบบแนวโค้งนูนขึ้น';
    showBanner(80,70,ShapeSTR,1,11);
    ShapeSTR:=#29#25'^PY-แสดงอักขระแบบแนวโค้งนูนลง';
    showBanner(80,90,ShapeSTR,1,6);
    ShapeSTR:=#30#23'^PW-แสดงข้อความแบบการเคลื่อนที่ของคลื่น';
    showBanner(0,200,ShapeSTR,1,7);
    ShapeSTR:=#31'^PT-ตัวยก'#20'กำลัง'#20'^PV-ตัว'#22'ห้อยท้าย';
    showBanner(90,310,ShapeSTR,1,11);
    readchr;
end;
```

รูปที่ 9.5 ชุดคำสั่งแสดงข้อความแบบโค้งและแบบคลื่น



รูปที่ 9.6 *ภาพที่ลอกจากจอภาพของชุดคำสั่งในรูปที่ 9.5*

ตามที่ได้แสดงโปรแกรมมานี้ ผู้อ่านจะเห็นว่าสามารถแสดงข้อความได้หลายแบบ โดยโปรแกรมหลักในการกำหนดให้ชุดคำสั่งที่ผ่านมาทำงานจะเป็นไปดังรูปที่ ๙.8 ซึ่งจะต้องกำหนดหน่วยความจำฮีปให้แก่ตัวแปรต่าง ๆ ด้วยก่อนจะเข้าไปรันโปรแกรมและแสดงข้อความตามที่ได้เขียนเป็นตัวอย่าง และก่อนที่จะจบโปรแกรมก็ต้องลบหน่วยความจำฮีปที่ใช้งานด้วย

อย่าลืมว่า...ท่านยังสามารถนำโปรแกรมนี้ไปแสดงผลแบบอื่น ๆ ได้อีกมากมาย ดังที่ปรากฏในโปรแกรม Thai Banner ที่จำหน่ายรวมกับคู่มือเล่มนี้ จึงฝากไว้ให้ลองคิดพัฒนาเพิ่มเติมเพื่อให้สามารถแสดงผลให้ตรงกับความต้องการต่อไป

```
{=================== ป้ายประกาศ 2 ======================}
begin
opengraph;
   Colorbg:=14; ColorBox:=14;     wavepoint;
   if Monoscreen then begin Colorbg:=1; ColorBox:=1; end;
   for font:=1 to 10 do
     begin
       Mark(Yscope[font]); new(Yscope[font]);
     end;
       LoadFile;PageNo:=1; NoSTR:=1;
       showContent1;
       showContent2;
       showContent3;
     for font:=1 to MaxFont do
```

```
    begin
      release(ShapePoly[font]);
      release(ShapeHead[font]);
    end;
   for font:=1 to 10 do  release (Yscope[font]);
  closegraph;
end.
```

รูปที่ 9.7 *โปรแกรมหลักในการทดสอบข้อความบนจอภาพ*

ฝากไว้ให้คิด

มีปัญหาที่ขอทิ้งไว้ให้ผู้อ่านลองคิดและเขียนโปรแกรมล่วงหน้าไปก่อนที่จะอ่านรายละเอียดในบทที่ 10 ดังนี้

- เขียนโปรแกรมเพื่อใช้ในการพิมพ์ภาพอย่างไร จึงจะเป็น Thai Banner ตามต้องการ
- เมื่อเป็นป้ายประกาศก็ต้องมีความยาวหลายหน้า ปัญหาคือ เราจะพิมพ์ข้อความยาว ๆ ลงในกระดาษได้อย่างไรเมื่อขนาดของจอภาพในการแสดงข้อความมีจำกัดเพียงขอบเขตของกราฟิกเท่านั้น
- ตำแหน่งเริ่มต้นของข้อความแต่ละข้อความ (ที่อาจมีหลายหน้า) จะทำอย่างไร

หวังว่าบางท่านคงจะได้แนวคิดในการพิมพ์ป้ายประกาศลงบนกระดาษบ้างพอสมควร และสามารถเขียนโปรแกรมทดลองเองล่วงหน้าได้ก่อนจะข้ามไปอ่านในบทที่ 10 ต่อไป

การสำเนาป้ายประกาศ 10

เมื่อได้สร้างป้ายประกาศที่สามารถแสดงผลบนจอภาพแล้ว ขั้นตอนต่อไปก็คือ การนำเอาป้ายประกาศที่ได้สร้างไปพิมพ์ลงกระดาษ โดยอาศัยวิธีการที่ง่ายที่สุดคือ การลอกหน้าจอไปลงกระดาษเลย ซึ่งวิธีนี้มีข้อดีคือถ้าคุณเห็นจอภาพอย่างไรก็จะได้ภาพแบบเดียวกัน ซึ่งก็คือลักษณะของโปรแกรมประเภท WYSIWYG นั่นเอง

สำหรับชุดคำสั่งเดิมที่จะต้องนำมาใช้ร่วมด้วยมีดังนี้

1. ส่วนเริ่มต้นของโปรแกรมและตัวแปรในการใช้งานทั้งหมด
2. LoadFile;
3. OpenGrpah;
4. ClsGr;
5. ReadChr;
6. SetPattern;
7. ScopePoint;
8. WavePoint;
9. ShowPoly;
10. ShowBanner;

จึงให้ทำการสำเนาชุดคำสั่งทั้งหมดที่กล่าวมาข้างต้นนี้ มาไว้ในโปรแกรมให้เป็นส่วนแรกเสียก่อนที่จะเขียนโปรแกรมในบทนี้ต่อ

การพิมพ์ลงกระดาษ

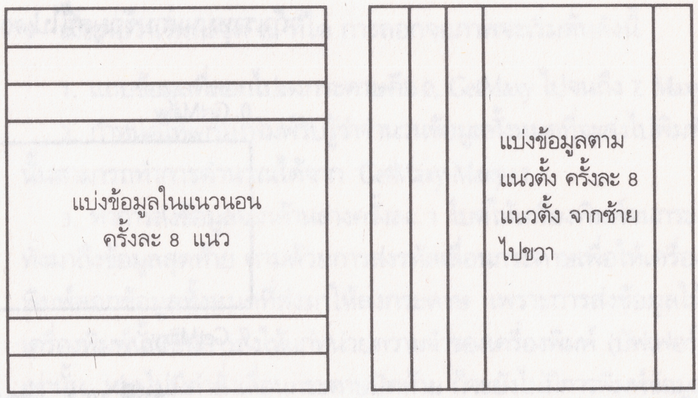
การลอกจอภาพในกรณีที่ทำเป็นป้ายประกาศนั้นจะใช้ชุดคำสั่งเหมือนการลอกจอภาพตามปกติไม่ได้ เพราะ

❖ โปรแกรมโดยทั่วไปจะเป็นการลอกจอภาพในแนวนอนเรียงจากบนลงล่าง ครั้งละ 8 แถว โดยส่งข้อมูลครั้งละ 8 บิตตามแนวตั้งจากซ้ายไปขวา (บนกระดาษจะเป็นการพิมพ์จากซ้ายไปขวาเช่นกัน) วิธีนี้จะได้ข้อความบนกระดาษในลักษณะเดียวกันกับจอภาพ

❖ พิมพ์ได้ครั้งละหนึ่งจอภาพ หากนำมาพิมพ์ป้ายประกาศก็คงต้องนำกระดาษแต่ละส่วนมาต่อให้เป็นแผ่นเดียวกันตามแนวตั้ง ซึ่งคงได้ป้ายประกาศที่ดูออกว่าเป็นการพิมพ์ทีละส่วน

❖ การพิมพ์ป้ายประกาศนั้นข้อความต้องยาว ดังนั้นการพิมพ์ที่เหมาะสมก็ต้องการให้พิมพ์ต่อเนื่องกันไปบนกระดาษต่อเนื่อง เพื่อให้มองเห็นว่าเป็นป้ายประกาศแผ่นเดียว โดยไม่มีการนำมาต่อเป็นแผ่นเดียวกันในภายหลัง

ดังนั้นเราจึงต้องลอกจอภาพอีกลักษณะหนึ่ง คือเปลี่ยนมาเป็นการลอกจอภาพในแนวตั้งเรียงจากซ้ายไปขวาครั้งละ 8 แนวตั้ง โดยสั่งพิมพ์ข้อมูลจากข้างล่างขึ้นข้างบน (บนกระดาษจะเป็นการพิมพ์จากซ้ายไปขวาเหมือนเดิม) จึงทำให้เป็นการพิมพ์ภาพจากจอภาพที่หมุนไปจากเดิม 90° โดยเปรียบเทียบการพิมพ์ทั้งสองแบบนี้ได้ดังรูปที่ 10.1



(ก) การลอกจอภาพตามปกติ (ข) การลอกจอภาพแบบหมุนภาพ 90°

รูปที่ 10.1 *เปรียบเทียบการลอกจอภาพลงกระดาษ 2 แบบ*

การสำเนาจอภาพลงกระดาษ

การลอกจอภาพเพื่อทำป้ายประกาศนี้เราจะใช้วิธีที่ 2 คือแบบ (ข) ในรูปที่ 10. โดยชุดคำสั่งจะทำงานดังนี้

❖ กำหนดระยะการเลื่อนกระดาษเป็นครั้งละ 24/216 นิ้ว (เครื่องพิมพ์แบบ 9 หัวเข็ม) หรือ 24/180 นิ้ว (เครื่องพิมพ์แบบ 24 หัวเข็ม)

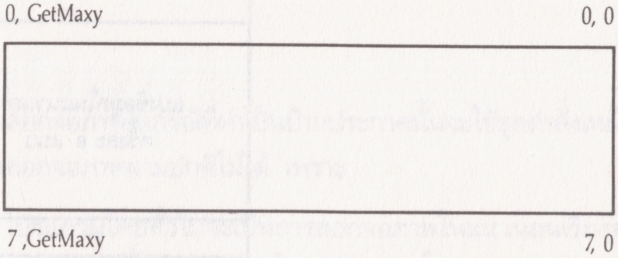
❖ กำหนดจุดเริ่มต้นการพิมพ์ ณ จุด 0, GetMaxy นั่นคือ จะเริ่มที่มุมล่างซ้าย

❖ กำหนดการลอกจอภาพครั้งละ 8 แนวตั้ง การสำเนาจอภาพแนวแรกจึงเป็นการกระทำกับข้อมูลของจอภาพในขอบเขต 0, Getmaxy ถึง 7, 0 เมื่อ

0,GetMaxy คือมุมล่างซ้ายของแถบข้อมูลที่จะลอกลงกระดาษ

7,0 คือมุมบนขวาของแถบข้อมูลที่จะลอกลงกระดาษ

ถ้าเราหมุนแถบข้อมูลนี้ไป ๙๐ ° จะได้ข้อมูลดังรูปที่ 10.2



รูปที่ 10.2 *แสดงการหมุนแถบข้อมูลของจอภาพ แถบแรกไป ๙๐ ° (ตามเข็มนาฬิกา)*

❖ เพื่อให้การพิมพ์ข้อมูลลงกระดาษมีความไวมากขึ้น เราจึงเริ่มต้นด้วยการตรวจแถบข้อมูลที่จะพิมพ์จากด้านบนลงล่าง (จาก ๐ ถึง GetMaxy) ครั้งละ ๘ บิต จึงได้ขนาดของข้อมูลเพียงหนึ่งไบต์ โดยเรากำหนดขนาดของข้อมูลเป็นแถวละหนึ่งบิต (แถวแรกแนวตั้งจะเป็นบิตที่ ๗ และแถวที่ ๘ แนวตั้งคือบิตที่ ๐)

การตรวจข้อมูลจะหยุดการตรวจเมื่อพบว่าบิตใดบิตหนึ่งไม่เป็น ๐ (คือมีการลงจุด) อาจมีผู้อ่านบางท่านคิดว่าหากมาตรวจแบบนี้น่าจะทำให้การพิมพ์ช้าลง เพราะต้องมาเสียเวลาคำนวณในเรื่องนี้เพิ่มเติมอีก จึงขอเรียนยืนยันว่าการทำอย่างนี้จะพิมพ์ภาพได้ไวกว่าอย่างแน่นอน ทั้งนี้เพราะว่าเครื่องคอมพิวเตอร์จะใช้เวลาที่รอการพิมพ์ที่แสนช้าของเครื่องพิมพ์มาทำการคำนวณประกอบด้วย

สมมติขณะที่หยุดตรวจแถบข้อมูลนั้นได้หยุดตรวจเมื่อนับจากบนมาล่างและได้ตัวแปร Maxy

❖ เมื่อพบว่าเริ่มลงจุด ณ ที่ใด การลอกจอภาพจะเริ่มต้นดังนี้

1. แถบข้อมูลที่ลอกไปลงกระดาษคือ 0, GetMaxy ไปจนถึง 7, Maxy

2. กำหนดให้เครื่องพิมพ์รับรู้ว่าจำนวนข้อมูลทั้งหมดที่จะส่งไปพิมพ์นั้นสามารถทำการคำนวณได้จาก GetMaxy-Maxy+1

3. ทำการส่งข้อมูลจากด้านล่างครั้งละ 1 ไบต์ให้เครื่องพิมพ์จนกระทั่งมาถึงข้อมูลสุดท้าย ตามด้วยการส่งรหัสเลื่อนกระดาษเพื่อให้เครื่องพิมพ์ลอกข้อมูลทั้งหมดที่ส่งมาให้ลงกระดาษ เพราะการส่งข้อมูลให้เครื่องพิมพ์นั้นเป็นการส่งให้แก่หน่วยความจำของเครื่องพิมพ์ (บัฟเฟอร์) เท่านั้น หากไม่มีคำสั่งเลื่อนกระดาษปิดท้าย ก็จะยังไม่มีการพิมพ์ข้อมูลนั้น ๆ ลงกระดาษ

เนื่องจากขนาดของข้อมูลในแต่ละแถบมีค่าได้สูงสุดไม่เกิน 350 จุด ดังนั้นเพื่อให้การพิมพ์เหมาะกับกระดาษขนาด 8 นิ้ว จึงเลือกใช้การพิมพ์กราฟิกแบบที่ 6 ซึ่งจะพิมพ์ได้สูงสุดถึง 720 จุด จึงต้องอ่านข้อมูล 1 ครั้ง แต่จะสั่งพิมพ์ 2 ครั้ง ทำให้ได้ภาพลงกระดาษขนาด 350X2 เท่ากับ 700 จุด

4. เลื่อนการตรวจแถบข้อมูลไปทางขวาครั้งละ 8 แถว ดังนั้นแถบที่ 2 จึงเป็นช่วง 7, GetMaxy ถึง 15, 0 แล้วกระทำตามลำดับที่ 3-5 ใหม่จนกว่าจะเลื่อนแถบข้อมูลมาเป็นแถบสุดท้าย ขวามือก็จบการสำเนาจอภาพ

❖ เลื่อนหัวพิมพ์ของเครื่องพิมพ์ให้มาอยู่ที่ด้านซ้ายสุดเพื่อเตรียมรับข้อมูลในหน้าถัดไป โดยไม่มีการใช้คำสั่งเลื่อนกระดาษทั้งหน้า (formfeed) ทั้งนี้เพื่อให้ข้อมูลที่พิมพ์นั้นเป็นข้อมูลที่พิมพ์ได้ต่อเนื่องเป็นแผ่นเดียวกัน

ชุดคำสั่งในการสำเนาจอภาพลงกระดาษตามแนวคิดนี้จะเป็นดังรูปที่ 10.3

```
PROCEDURE CopyByte(ByteSt :string);
var  B:byte;
begin
   for b:=1 to length(ByteSt) do
     begin
       Regs.AH := 0;      Regs.AL := ord(ByteSt[b]);;
       Regs.DX := Pred(1);  Intr($17, Regs);
     end;
end;

PROCEDURE HardCopy;
var Scanline,n1,n2:word; ColorOld:byte;
   PROCEDURE CopyDot;
      const
         Bits : array[0..7] of byte = (128,64,32,16,8,4,2,1);
      var
         bit,pixel:word; MaxY,XYPixel: integer;
      begin
         MaxY:=-1;
         Repeat
              inc(MaxY); Pixel:=0;
              for bit := 0 to 7 do
                    if (GetPixel(Scanline+bit,MaxY) <> 0) then
                           Pixel:= Pixel+ Bits[bit];
         until (Pixel<>0) or (MaxY=GetMaxY);
```

```
      if MaxY<>getMaxY then begin
              n1 := lo(succ(GetMaxY-MaxY+1)*2);
              n2 := Hi(succ(GetMaxY-MaxY+1)*2);
           copyByte(#27'3'#24#27'*'#6+chr(n1)+chr(n2));
            for XyPixel:=getMaxY downto MaxY-1 do
              begin  Pixel := 0;
                  for bit := 0 to 7 do
                     if  GetPixel(Scanline+bit,XyPixel) <> 0 then
                               Pixel := Pixel+ Bits[bit];
                     CopyByte(chr(Pixel)+chr(Pixel));
              end;            end;
      CopyByte(#10);
      end;

begin
      CopyByte(#27'1'#3);  ScanLine:=0;
      Repeat
         CopyDot;
         Inc(Scanline,8);
      until ScanLine>GetMaxX;
      Copybyte(#27#2#27'1'#0#27'<');
end;
```

รูปที่ 10.3 ชุดคำสั่งในการสำเนาจอภาพลงกระดาษแบบหมุนภาพไป 90 ̊

เทคนิคการพิมพ์ป้ายประกาศ

เนื่องจากข้อความที่เราพิมพ์จากป้ายประกาศนั้น มีความยาวมากกว่าที่จะแสดงให้จบได้ในหนึ่งจอภาพ เช่น อาจมีถึง 4 จอภาพ 6 จอภาพเป็นต้น ดังนั้นสิ่งสำคัญในการสำเนาจอภาพจึงไม่ได้มีเพียงการพิมพ์แบบหมุนไป 90° เท่านั้น เรายังต้องกำหนดในเรื่องอื่น ๆ ด้วย ถึงจะสามารถพิมพ์ข้อความทั้งหมดลงในกระดาษต่อเนื่องได้อย่างสวยงาม

เทคนิคที่ผู้เขียนใช้นั้นอาศัยที่ภาษาปาสคาลได้กำหนดรูปแบบของคำสั่งที่เหมาะสมมาให้แล้ว คือคำสั่งที่เกี่ยวกับการวาดภาพกราฟิกนั้นได้กำหนดข้อมูลเป็นชนิด integer เช่น

```
line(x1,y1,x2,y2:integer);
circle(x,y:integer; Radius:word);
```

ซึ่งทำให้เราสามารถกำหนดค่าของตัวแปรเหล่านี้เป็นค่าลบได้ (คือให้แสดงก่อนถึงขอบซ้ายสุดของจอภาพ) จึงนำส่วนที่ภาษาปาสคาลเอื้ออำนวยให้นี้มากำหนดเทคนิคการลอกจอภาพให้เป็นป้ายประกาศได้

เพื่อให้การใช้เทคนิคต่อไปนี้มีความสมบูรณ์ จึงจะนำตัวแปร PageNo มาใช้งานร่วมด้วย (ในบทที่ ๑ กำหนดให้แสดงเพียงหน้าเดียว จึงกำหนด PageNo =1)

สิ่งที่เราต้องไม่ลืมก็คือ โปรแกรมนี้จะต้องรันได้ทั้งจอโมโนโครมและจอสี ซึ่งทั้งสองจอนี้จะมีขอบเขตของจอภาพไม่เท่ากัน ดังนั้นการสำเนาจอภาพจึงจะต้องคำนึงถึงขอบเขตของจอภาพเพื่อที่จะได้ป้ายประกาศที่ทำงานถูกต้องได้ทุกจอภาพ

เทคนิคที่ใช้มีดังนี้

❖ กำหนดตัวแปรเพื่อเก็บข้อมูลและข้อความไว้ดังนี้

1. ข้อมูลถูกกำหนดในตัวแปรข้อมูลมิติชื่อ BannerData เป็นตัวแปรแบบระเบียนข้อมูลที่มี 5 เขตข้อมูลคือ

BannerData[].x เก็บจุดเริ่มต้นของข้อความตามแกน x โดยจะให้ค่าเท่าใดก็ได้ เช่น

ให้ค่า 300 ก็จะแสดงในหน้าแรก ให้ค่าเป็น 800 ก็จะไปแสดงในหน้าที่สอง เป็นต้น

BannerData[].y เก็บจุดเริ่มต้นของข้อความในแกน y มีค่าตามขอบเขตของจอภาพคือ 0-350

BannerData[].str เก็บข้อความที่ต้องการทำป้ายประกาศ ซึ่งรวมทั้งรหัสการแสดงผลของข้อความแบบต่าง ๆ ด้วย

BannerData[].size เก็บขนาดของข้อความ

BannerData[].patt เก็บรูปแบบพื้นของข้อความ

2. จำนวนข้อมูลกำหนดไว้ให้ไม่เกิน 10 ข้อมูล

❖ กำหนดตัวแปรอื่น ๆ ที่เกี่ยวข้อง เช่น

Max คือจำนวนสูงสุดของข้อความที่ต้องการแสดงบนป้ายประกาศ

MaxPage คือจำนวนหน้าสูงสุด (ความยาวของป้ายประกาศ) ที่จะแสดงข้อความของป้ายประกาศ ค่าที่ผู้อ่านจะต้องรันดูเองว่าได้รูปของป้ายประกาศบนจอภาพครบถ้วนหรือไม่ แล้วกำหนดให้ตรงกับขนาดของป้ายประกาศที่ท่านกำหนดไว้

PageNo คือหมายเลขหน้าของป้ายประกาศ ซึ่งจะหมายถึงเป็นจอภาพลำดับที่เท่าใด โดยเริ่มที่ PageNo = 1 เสมอ

❖ เทคนิคสำคัญในการนำข้อความที่กำหนดไว้มาแสดงบนจอภาพมีดังนี้

1. เริ่มต้นหน้าแรกด้วยค่า Bannerdata[].x เท่าที่กำหนดไว้

2. แสดงข้อความทั้งหมดบนจอภาพ โดยแต่ละข้อความจะหยุดการแสดงถ้าข้อความนั้น ๆ มีขอบเขตเกินด้านขวาสุด (GetMaxx) ของจอภาพที่ใช้

3. เมื่อจบการแสดงข้อความแล้ว ค่า BannerData[1].x นี้จะเปลี่ยนค่าไปจากเดิม คือลดลงครั้งละ GetMaxx ด้วยคำสั่ง

BannerData[].x:=BannerData[].x-GetMaxx+1;

ตัวอย่างเช่น เดิมข้อมูลเป็น BannerData[].x:=800; ซึ่งถ้าเป็นหน้าแรกข้อความนี้จะไม่แสดง (เพราะค่า 800 มากกว่า GetMaxx) แต่เมื่อมาแสดงที่หน้าที่สอง ค่านี้จะเปลี่ยนเป็น BannerData[].x=800-639+1=162 (บนจอสี) ซึ่งถ้าให้เริ่มในขอบเขตของจอภาพจึงจะเริ่มแสดงข้อความนี้ในหน้าที่สองเป็นต้นไป

4. รอกดแป้นเพื่อดูผลของป้ายประกาศบนจอภาพ หากต้องการลอกลงกระดาษก็ให้กดแป้น [P] หรือ [p] ซึ่งจะเรียกใช้ชุดคำสั่ง Hardcopy มาทำงาน หากกดแป้นอื่นก็จะเป็นการเลื่อนไปยังหน้าถัดไป

5. เพิ่มค่าตัวแปร PageNo ครั้งละ 1

6. วนรอบกลับไปทำข้อ (2) จนกว่าจะจบหน้า (Maxpage) ที่กำหนด

การทำตามเทคนิคนี้ก็จะได้ป้ายประกาศที่พิมพ์บนกระดาษต่อเนื่องได้อย่างสวยงาม ตัวอย่างการพิมพ์ป้ายประกาศจะเป็นดังรูปที่ 10.4

```
PROCEDURE Show_Banner;
var k,no:word; MaxPage,max:byte;
begin
  BannerData[1].STR:=#29#12การใช้คอมพิวเตอร์ให้คุ้มค่ามากยิ่งขึ้น';
  BannerData[1].x:=60;  BannerData[1].y:=220;
  BannerData[1].Size:=5;  BannerData[1].patt:=19;
  BannerData[2].STR:=#28#23โปรแกรมป้ายประกาศ โดยอาจหาญ สัตยารักษ์';
```

```
   BannerData[2].x:=400; BannerData[2].y:=300;
   BannerData[2].Size:=2; BannerData[2].patt:=16;
   BannerData[3].STR:=#28#5#25'ได้รูปแบบของป้ายประกาศที่เป็นของตนเอง';
   BannerData[3].x:=200; BannerData[3].y:=30;
   BannerData[3].Size:=1; BannerData[3].patt:=1;
   BannerData[4].STR:=#28#12#19#5'น่าใช้  '#5#12'อย่าลืม...สอบถามได้
   BannerData[4].x:=1500;BannerData[4].y:=56;
   BannerData[4].Size:=2; BannerData[4].patt:=0;
   BannerData[5].STR:=#29#19#21'สนใจหรือเปล่า!!';
   BannerData[5].x:=2400;BannerData[5].y:=320;
   BannerData[5].Size:=2; BannerData[5].patt:=20;
   Max:=5; MaxPage:=5;  PageNo:=1;
 for k:=1 to maxpage do
   begin
     for no:=1 to Max do
       begin
         NoSTR:=no;
         showBanner(BannerData[no].x,BannerData[no].y,BannerData[no].STR,
                    BannerData[no].size,BannerData[no].Patt);
         BannerData[no].x:=BannerData[no].x-GetMaxX+1;
      end;
      readchr;  if Upcase(ch)='P' then  hardcopy;
      clsGR(0,0,GetMaxX,GetMaxY);  Inc(PageNo);
  end;
 end;
```

รูปที่ 10.4 *ตัวอย่างการกำหนดข้อความเพื่อทำป้ายประกาศ*

วิเคราะห์การพิมพ์ป้ายประกาศ

เราจะลองพิจารณาการทำงานของโปรแกรมในรูปที่ 10.4 เพื่อเป็นแนวทางในการเรียนรู้การเขียนโปรแกรม โดยขอยกตัวอย่างของข้อความหมายเลข 5 มาเป็นตัวอย่าง ดังนี้

❖ จุดเริ่มต้นคือ BannerData[5].x: =2400; และข้อความกำหนดขนาดไว้เป็นขนาดที่ 2 (BannerData[5].size: =2;)

❖ นับความยาวของข้อความที่ 5 โดยไม่นับสระ วรรณยุกต์ จะได้ทั้งหมด 13 ตัวอักษร ข้อความนี้จึงควรมีความยาวตามสูตรง่าย ๆ คือ

ความยาว = จำนวนอักขระที่นับได้ x ความกว้างอักขระ x ขนาดข้อความ
= 13 X 20 X 2
= 520 จุด

ค่าที่คำนวณได้นี้เป็นค่าโดยประมาณ เพราะว่าตัวอักขระของเรามีความกว้างไม่เท่ากัน แต่เราจะใช้ความกว้างสูงสุดมาคำนวณ

❖ เมื่อ PageNo =1 แต่ค่า BannerData[5].x>639 ดังนั้น จึงไม่แสดงข้อความนี้

❖ เมื่อ PageNo = 2 ค่า BannerData[5].x = 2400-639+1=1762>639 จึงยังไม่แสดงข้อความนี้เช่นเดิม

❖ เมื่อ PageNo = 3 ค่า BannerData[5].x = 1762-639+1=1124>639 จึงยังไม่แสดงข้อความนี้เช่นเดิม

❖ เมื่อ PageNo = 4 ค่า BannerData[5].x = 1124-639+1=486<639 จึงเริ่มแสดงข้อความนี้ที่หน้าที่ 4 เมื่อหาส่วนท้ายสุดของข้อความนี้จะอยู่ที่ตำแหน่ง (639-486)+1+520 = 674 ดังนั้นข้อความนี้จึงจะต้องไปอยู่ในหน้า 5 ด้วย

❖ เมื่อ PageNo = 5 ค่า BannerData[5].x = 486-639+1 = -152<0 ข้อความนี้จะแสดงที่หน้านี้ แต่จะเป็นการแสดงต่อเนื่องจากหน้าที่ 4 เพราะข้อความส่วนหนึ่งจะแสดงที่นอกขอบซ้ายสุด (ค่าเป็นลบ)

การคำนวณหาจำนวนหน้าสูงสุด

ผู้อ่านอาจเพิ่มชุดคำสั่งเพื่อคำนวณหาจำนวนหน้าสูงสุด (MaxPage) เพิ่มเติมได้ โดยใช้หลักการดังนี้

❖ หาจุดขวาสุดของแต่ละข้อความจากสูตรง่าย ๆ ดังนี้

จุดขวาสุด = จุดเริ่มต้น + จำนวนอักขระที่นับได้ x 20 x ขนาด

ตามตัวอย่างจะได้ดังนี้

จุดขวาสุด(1) = 60 + 27X20X5 = 2760

จุดขวาสุด(2) = 400 + 34X20X2 = 1760

จุดขวาสุด(3) = 200 + 31X20X1 = 820

จุดขวาสุด(4) = 1500 + 24X20X2 = 2460

จุดขวาสุด(5) = 2400 + 3X20X2 = 2920

นั่นคือ จุดขวาสุดของข้อความทั้งหมดอยู่ที่จุด 2920 (ข้อความที่ 5)

❖ หาจำนวนหน้าสูงสุดได้จากสูตร

จำนวนหน้าสูงสุด = จุดขวาสุด/Getmaxx

= 2920/639

= 4.56964

= 5

❖ กำหนดค่าที่หาได้ในลำดับที่ 2 ให้กับตัวแปร MaxPage ก็จะได้ป้ายประกาศที่แสดงข้อความได้ครบถ้วนทุกข้อความ

โดยส่วนการคำนวณนี้ ผู้เขียนไม่ได้เขียนชุดคำสั่งไว้ให้ เพราะต้องการให้ผู้อ่านเขียนส่วนนี้เพิ่มเติมเอง ซึ่งคงไม่ยากเกินกว่าที่จะเขียนได้เอง

❖ โปรแกรมหลักในการทำงานของป้ายประกาศ ตัวอย่างจะเป็นดังรูปที่ 10.5

```
{================ ป้ายประกาศ 3 ================}
begin
  opengraph;
   colorbg:=14; ColorBox:=14;
   if monoscreen then
    begin
      colorbg:=1; ColorBox:=1;
    end;
    for font:=1 to 10 do
     begin
       mark(Yscope[font]);
       new(Yscope[font]);
     end;
   LoadFile;  WavePoint;
   Show_Banner;
    for font:=1 to MaxFont do
     begin
       release(ShapePoly[font]);
       release(ShapeHead[font]);
     end;
    for font:=1 to 10 do  release(Yscope[font]);
  closegraph;
end.
```

รูปที่ 10.5 *โปรแกรมหลักในการพิมพ์ตัวอย่างป้ายประกาศ*

สำหรับการพิมพ์ตัวอย่างของป้ายประกาศนี้ ไม่สามารถทำได้ในขนาดเท่าของจริง จึงได้พิมพ์ป้ายประกาศมาแล้วถ่ายเอกสารแบบย่อขนาดลง โดยขนาดจริงของข้อความที่ว่าการใช้คอมพิวเตอร์ให้คุ้มค่ามากยิ่งขึ้นนั้น จะมีความสูงประมาณ 5 เซนติเมตร และป้ายประกาศยาวทั้งหมดประมาณ 1 เมตร ซึ่งขนาดที่ย่อแล้วจะเป็นดังรูปที่ 10.6

รูปที่ 10.6 *ตัวอย่างภาพป้ายประกาศตามข้อความในรูป 10.4*

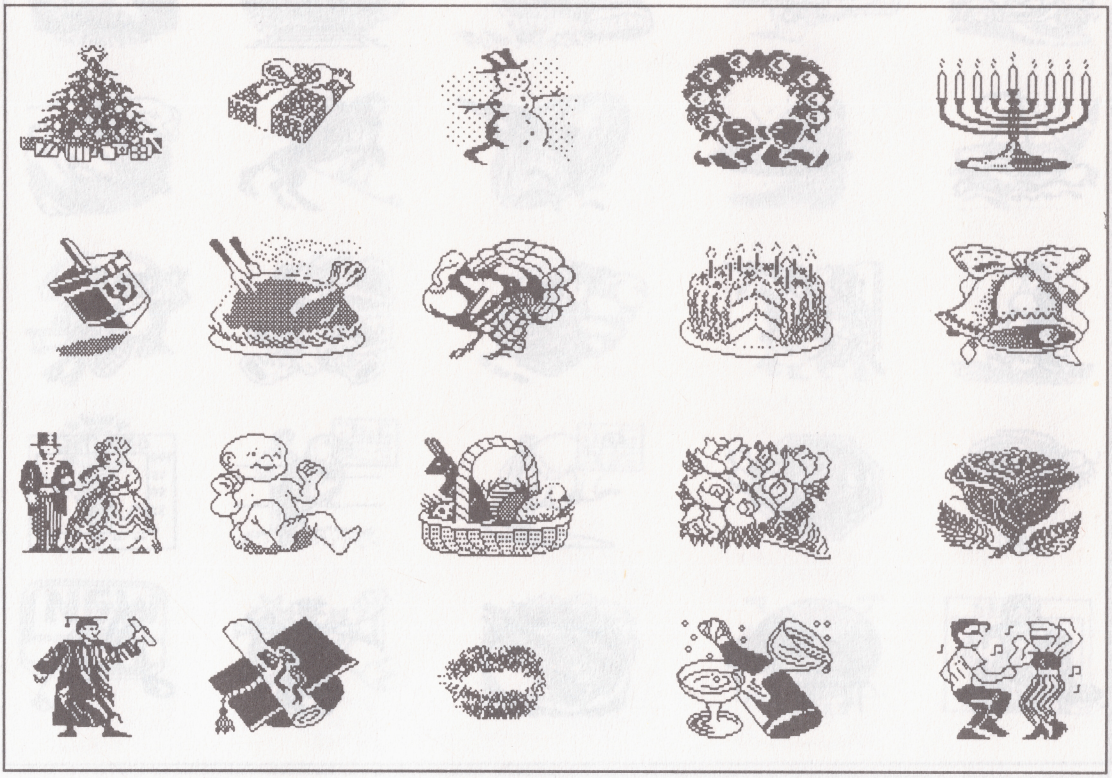
ผู้เขียนหวังว่าส่วนที่สองของหนังสือเล่มนี้ จะมีส่วนช่วยให้ท่านมองเห็นลู่ทางการพัฒนาโปรแกรมให้ตรงกับความต้องการของท่านมากยิ่งขึ้น ส่วนท่านที่ไม่มีความรู้ในเรื่องการเขียนโปรแกรม คงจะได้ใช้โปรแกรม Thai Banner ที่เป็นโปรแกรมสำเร็จเรียบร้อยแล้ว ที่จำหน่ายพร้อมกับหนังสือเล่มนี้

ผู้เขียนขอขอบคุณทุกท่านมาพร้อมกันนี้ด้วย

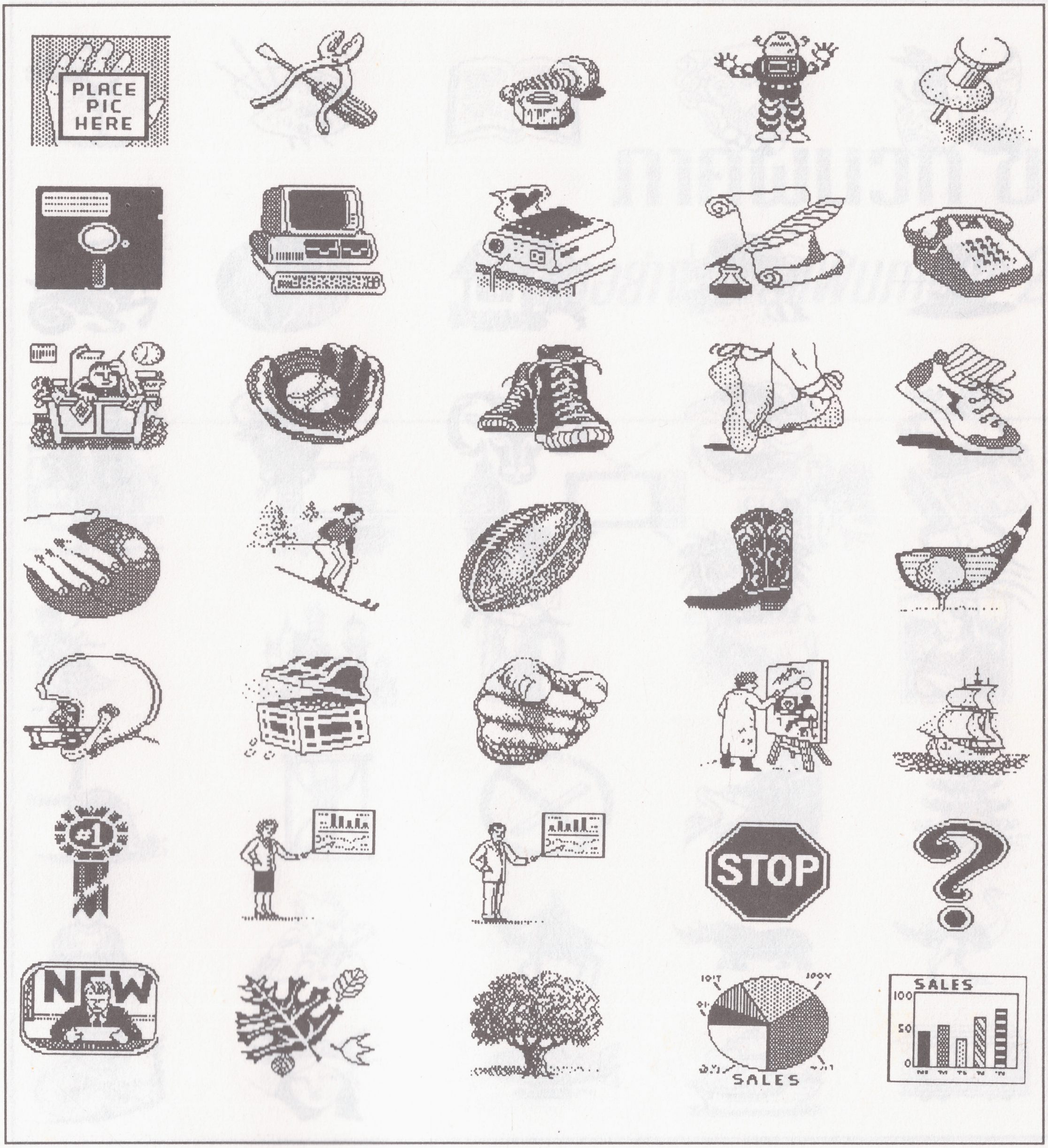
ขอให้โชคดีครับ...

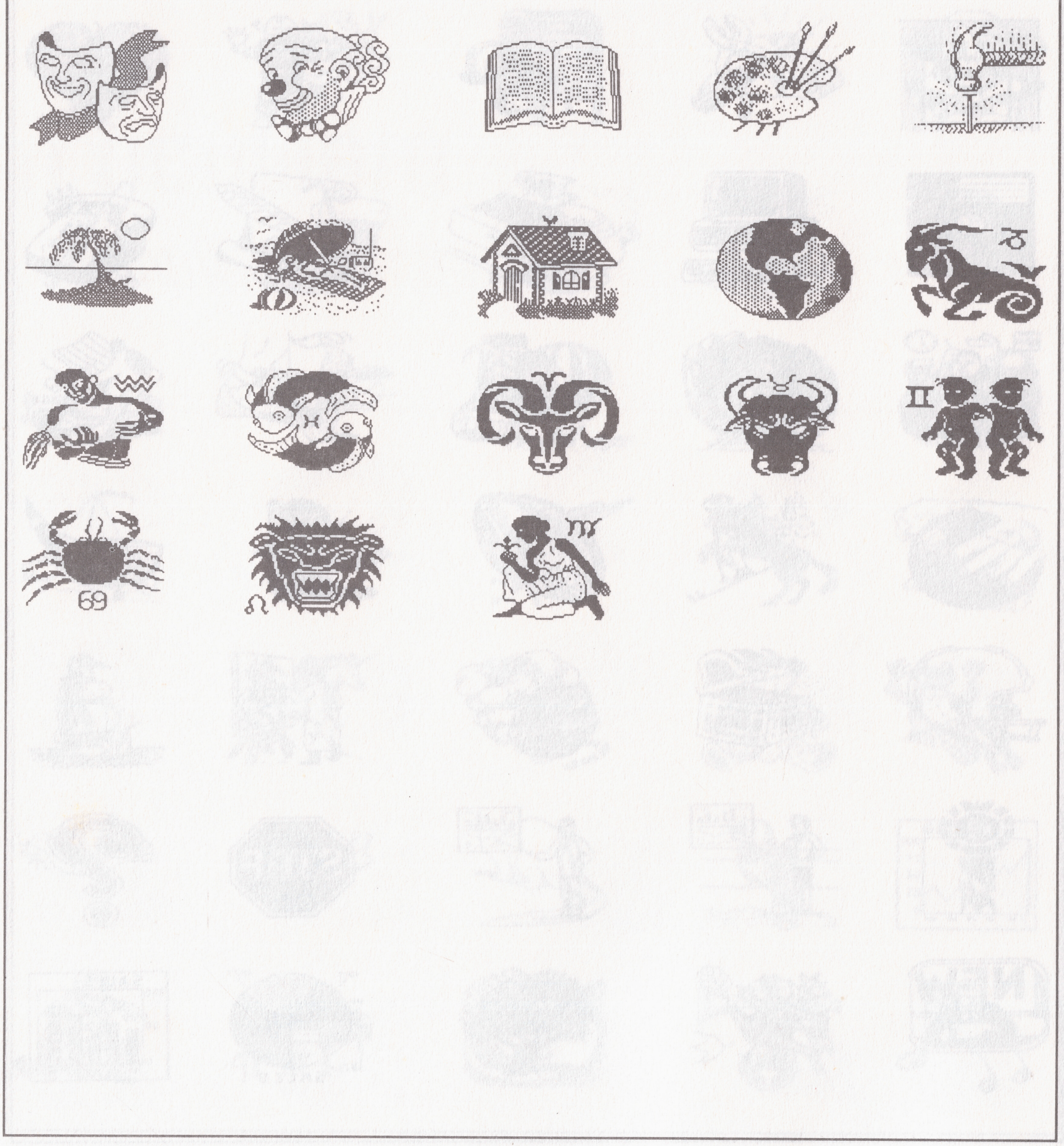
ภาคผนวก ก

ตัวอย่างรูปในแฟ้มภาพที่ 1



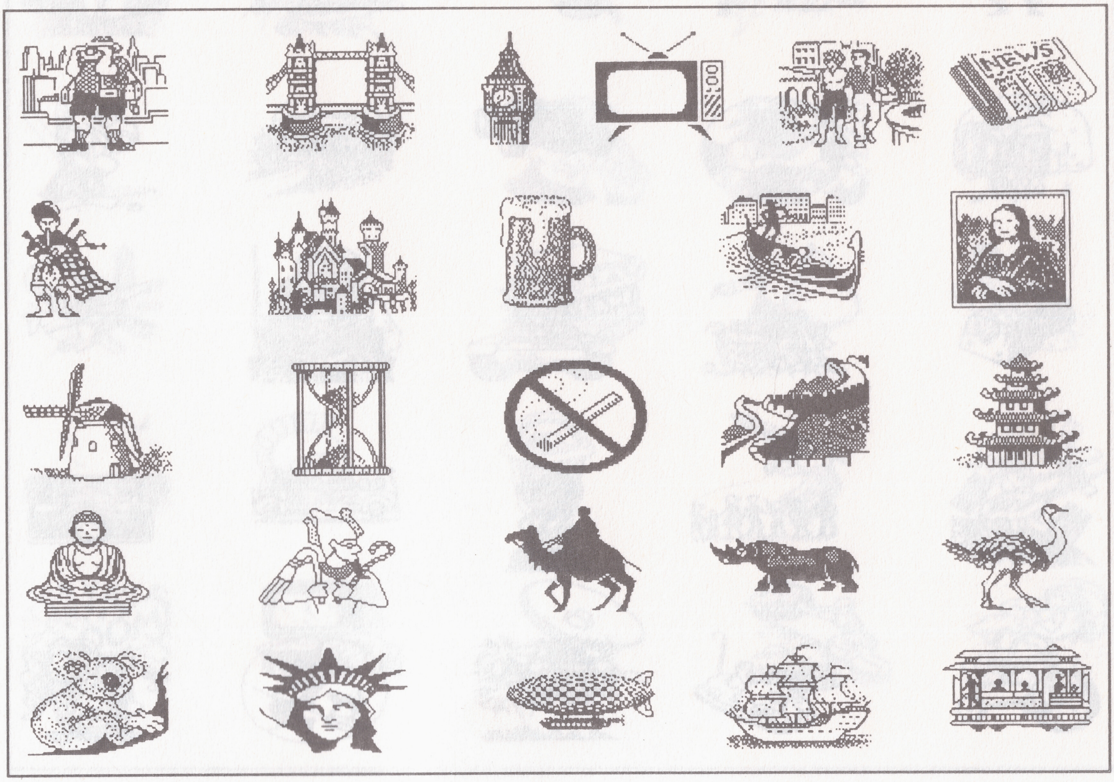
GROWTH
100
75
50
25
0


PLACE
PIC
HERE
#1
STOP
NEW
SALES
SALES
100
50
0



ภาคผนวก ข

ตัวอย่างรูปในแฟ้มภาพที่ 2



SCHOOL

BOOM
BOOM
LAW
LAW
BOOM

จะมีประโยชน์อะไร
ถ้ามีคอมพิวเตอร์แล้วไม่รู้จะใช้มันอย่างไรดี !

นับแต่นี้ไป ผู้ใช้คอมพิวเตอร์และผู้ที่ต้องเกี่ยวข้องกับคอมพิวเตอร์ทุกระดับ
จะได้พบกับนิตยสารรายเดือนคู่ใจ ในนาม **IT SOFTWARE**
ซึ่งเป็นนิตยสารคุณภาพอีกฉบับหนึ่งในเครือบริษัท อินฟอร์เมติก บิซิเนส พับลิเคชัน จำกัด (IBIZ)

IT SOFTWARE ช่วยแยกแยะเรื่องราวต่าง ๆ ทางด้านซอฟต์แวร์และด้านความก้าวหน้าพร้อมการใช้งานเพื่อท่านผู้อ่าน ประกอบด้วยบทความทางเทคโนโลยีซอฟต์แวร์ ความเคลื่อนไหวในวงการซอฟต์แวร์เมืองไทยและต่างประเทศ เลือกซื้อ

เลือกใช้ซอฟต์แวร์ที่เหมาะสมกับงานในรูปแบบต่าง ๆ และบทความแนวเทคนิคการใช้งาน เกร็ดซอฟต์แวร์ต่าง ๆ **IT SOFTWARE** คือ คำตอบในส่วนที่ขาดหายไปของผู้ใช้คอมพิวเตอร์ทุกระดับ ติดตามได้ทุกต้นเดือน